Informm℠

NATIONAL Intsruments

THE NASDAQ STOCK MARKET℠

NASDAQ®

LAB VIEW

LabVIEW™

Getting Started with LabVIEW

NATIONAL INSTRUMENTS™

April 2003 Edition
Part Number 323427A-01

Digitized by the Internet Archive
in 2025

https://archive.org/details/manualsplus_06881

Worldwide Technical Support and Product Information

ni.com

National Instruments Corporate Headquarters

11500 North Mopac Expressway Austin, Texas 78759-3504 USA Tel: 512 683 0100

Worldwide Offices

Australia 61 2 9672 8846, Austria 43 0 662 45 79 90 0, Belgium 32 0 2 757 00 20, Brazil 55 11 3262 3599,
Canada (Calgary) 403 274 9391, Canada (Montreal) 514 288 5722, Canada (Ottawa) 613 233 5949,
Canada (Québec) 514 694 8521, Canada (Toronto) 905 785 0085, Canada (Vancouver) 514 685 7530,
China 86 21 6555 7838, Czech Republic 420 2 2423 5774, Denmark 45 45 76 26 00,
Finland 385 0 9 725 725 11, France 33 0 1 48 14 24 24, Germany 49 0 89 741 31 30, Greece 30 2 10 42 96 427,
Hong Kong 2645 3186, India 91 80 51190000, Israel 972 0 3 6393737, Italy 39 02 413091,
Japan 81 3 5472 2970, Korea 82 02 3451 3400, Malaysia 603 9059 6711, Mexico 001 800 010 0793,
Netherlands 31 0 348 433 466, New Zealand 64 09 914 0488, Norway 47 0 32 27 73 00,
Poland 48 0 22 3390 150, Portugal 351 210 311 210, Russia 7 095 238 7139, Singapore 65 6 226 5886,
Slovenia 386 3 425 4200, South Africa 27 0 11 805 8197, Spain 34 91 640 0085, Sweden 46 0 8 587 895 00,
Switzerland 41 56 200 51 51, Taiwan 886 2 2528 7227, United Kingdom 44 0 1635 523545

For further support information, refer to the *Technical Support and Professional Services* appendix. To comment on the documentation, send email to techpubs@ni.com.

© 2003 National Instruments Corporation. All rights reserved.

Important Information

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

Trademarks

CVI™, DAQPad™, LabVIEW™, Measurement Studio™, National Instruments™, NI™, NI Developer Zone™, ni.com™, NI-DAQ™, and SCXI™ are trademarks of National Instruments Corporation.

FireWire® is the trademark of Apple Computer, Inc., registered in the U.S. and other countries. Other product and company names mentioned herein are trademarks or trade names of their respective companies.

Patents

For patents covering National Instruments products, refer to the appropriate location: **Help»Patents** in your software, the `patents.txt` file on your CD, or `ni.com/patents`.

WARNING REGARDING USE OF NATIONAL INSTRUMENTS PRODUCTS

(1) NATIONAL INSTRUMENTS PRODUCTS ARE NOT DESIGNED WITH COMPONENTS AND TESTING FOR A LEVEL OF RELIABILITY SUITABLE FOR USE IN OR IN CONNECTION WITH SURGICAL IMPLANTS OR AS CRITICAL COMPONENTS IN ANY LIFE SUPPORT SYSTEMS WHOSE FAILURE TO PERFORM CAN REASONABLY BE EXPECTED TO CAUSE SIGNIFICANT INJURY TO A HUMAN.

(2) IN ANY APPLICATION, INCLUDING THE ABOVE, RELIABILITY OF OPERATION OF THE SOFTWARE PRODUCTS CAN BE IMPAIRED BY ADVERSE FACTORS, INCLUDING BUT NOT LIMITED TO FLUCTUATIONS IN ELECTRICAL POWER SUPPLY, COMPUTER HARDWARE MALFUNCTIONS, COMPUTER OPERATING SYSTEM SOFTWARE FITNESS, FITNESS OF COMPILERS AND DEVELOPMENT SOFTWARE USED TO DEVELOP AN APPLICATION, INSTALLATION ERRORS, SOFTWARE AND HARDWARE COMPATIBILITY PROBLEMS, MALFUNCTIONS OR FAILURES OF ELECTRONIC MONITORING OR CONTROL DEVICES, TRANSIENT FAILURES OF ELECTRONIC SYSTEMS (HARDWARE AND/OR SOFTWARE), UNANTICIPATED USES OR MISUSES, OR ERRORS ON THE PART OF THE USER OR APPLICATIONS DESIGNER (ADVERSE FACTORS SUCH AS THESE ARE HEREAFTER COLLECTIVELY TERMED "SYSTEM FAILURES"). ANY APPLICATION WHERE A SYSTEM FAILURE WOULD CREATE A RISK OF HARM TO PROPERTY OR PERSONS (INCLUDING THE RISK OF BODILY INJURY AND DEATH) SHOULD NOT BE RELIANT SOLELY UPON ONE FORM OF ELECTRONIC SYSTEM DUE TO THE RISK OF SYSTEM FAILURE. TO AVOID DAMAGE, INJURY, OR DEATH, THE USER OR APPLICATION DESIGNER MUST TAKE REASONABLY PRUDENT STEPS TO PROTECT AGAINST SYSTEM FAILURES, INCLUDING BUT NOT LIMITED TO BACK-UP OR SHUT DOWN MECHANISMS. BECAUSE EACH END-USER SYSTEM IS CUSTOMIZED AND DIFFERS FROM NATIONAL INSTRUMENTS' TESTING PLATFORMS AND BECAUSE A USER OR APPLICATION DESIGNER MAY USE NATIONAL INSTRUMENTS PRODUCTS IN COMBINATION WITH OTHER PRODUCTS IN A MANNER NOT EVALUATED OR CONTEMPLATED BY NATIONAL INSTRUMENTS, THE USER OR APPLICATION DESIGNER IS ULTIMATELY RESPONSIBLE FOR VERIFYING AND VALIDATING THE SUITABILITY OF NATIONAL INSTRUMENTS PRODUCTS WHENEVER NATIONAL INSTRUMENTS PRODUCTS ARE INCORPORATED IN A SYSTEM OR APPLICATION, INCLUDING, WITHOUT LIMITATION, THE APPROPRIATE DESIGN, PROCESS AND SAFETY LEVEL OF SUCH SYSTEM OR APPLICATION.

Contents

About This Manual

Conventions ix

Chapter 1
Getting Started with LabVIEW Virtual Instruments

Building a Virtual Instrument 1-1
- Opening a New VI from a Template 1-2
- Adding a Control to the Front Panel 1-4
- Changing the Signal Type 1-5
- Wiring Objects on the Block Diagram 1-7
- Running the VI 1-8
- Modifying the Signal 1-9
- Displaying Two Signals on the Graph 1-11
- Customizing the Knob 1-12
- Customizing the Waveform Graph 1-14

Summary 1-16
- New Dialog Box and VI Templates 1-16
- Front Panel 1-16
- Property Dialog Boxes 1-16
- Block Diagram 1-17
- Express VIs 1-17

Chapter 2
Analyzing and Saving a Signal

Building a VI from a Template 2-1
- Opening a New VI from a Template 2-2
- Modifying the Block Diagram 2-3
- Modifying the Front Panel 2-4
- Analyzing the Amplitude of a Signal 2-5
- Adding a Warning Light 2-6
- Setting the Warning Level Limit 2-7
- Warning the User 2-8
- Configuring the VI to Save Data to a File 2-9
- Saving Data to a File 2-9
- Adding a Button that Stores Data when Pressed 2-10
- Saving Data when Prompted by a User 2-10

Summary 2-13
- LabVIEW Help Resources 2-13

Controls and Indicators 2-13
Saving Data 2-14
Errors and Broken Wires 2-14

Chapter 3
Expanding the Features of a VI

Building a VI from a Blank Template 3-1
Opening a Blank VI 3-2
Adding an Express VI that Simulates a Signal 3-2
Modifying the Signal 3-3
Customizing the Front Panel 3-4
Configuring the VI to Run Continuously Until the User Stops It 3-5
Controlling the Speed of Execution 3-6
Using a Table to Display Data 3-7
Summary 3-9
Using the LabVIEW Help Resources 3-9
Customizing the Block Diagram Code 3-9
Creating Controls and Indicators 3-9
Controlling When a VI Stops Running 3-10
Displaying Data in a Table 3-10

Chapter 4
Acquiring Data and Communicating with Instruments

Acquiring a Signal 4-1
Creating an NI-DAQmx Task 4-2
Testing the Task 4-4
Graphing Data from a DAQ Device 4-4
Editing an NI-DAQmx Task 4-5
Visually Comparing Two Voltage Readings 4-6
Communicating with an Instrument 4-6
Selecting an Instrument 4-6
Acquiring and Parsing Information for an Instrument 4-7
Summary 4-8
DAQ Assistant Express VI 4-8
Tasks 4-8
Instrument I/O Assistant Express VI 4-9

Chapter 5
Using Other LabVIEW Features

NI Example Finder....5-1
All Controls and Indicators....5-1
All VIs and Functions....5-2
VIs....5-3
Functions....5-3
Data Types....5-4
Dynamic Data Type....5-4
Converting from Dynamic Data....5-4
Converting to Dynamic Data....5-5
When to Use Other LabVIEW Features....5-6

Appendix A
Technical Support and Professional Services

Glossary

Index

About This Manual

Use this manual to familiarize yourself with the LabVIEW graphical programming environment and the basic LabVIEW features you use to build data acquisition and instrument control applications.

This manual contains exercises to teach you how to develop basic applications in LabVIEW. These exercises take a short amount of time to complete and help you get started with LabVIEW.

Throughout each exercise, you will see illustrations that provide information about the concepts associated with that step. The end of each chapter includes a summary of the main concepts taught in that chapter. Use these summaries to review what you learned.

In addition to this manual, LabVIEW includes the *LabVIEW User Manual*, *LabVIEW Help*, other reference manuals, Application Notes, and examples. If you select the **Complete** install option, LabVIEW installs PDF versions of all LabVIEW manuals, which you can access by selecting **Help»Search the LabVIEW Bookshelf** in LabVIEW.

Note You must have Adobe Acrobat Reader with Search and Accessibility 5.0.5 or later installed to view the PDFs. Refer to the Adobe Systems Incorporated Web site at `www.adobe.com` to download Acrobat Reader.

Conventions

The following conventions appear in this manual:

»

The » symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.

This icon denotes a tip, which alerts you to advisory information.

This icon denotes a note, which alerts you to important information.

bold

Bold text denotes items that you must select or click in the software, such as menu items and dialog box options. Bold text also denotes parameter names, controls and buttons on the front panel, dialog boxes, sections of dialog boxes, menu names, and palette names.

italic	Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept. This font also denotes text that is a placeholder for a word or value that you must supply.
monospace	Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames and extensions, and code excerpts.
monospace bold	Bold text in this font denotes the messages and responses that the computer automatically prints to the screen. This font also emphasizes lines of code that are different from the other examples.

1

Getting Started with LabVIEW Virtual Instruments

LabVIEW programs are called virtual instruments, or VIs, because their appearance and operation imitate physical instruments, such as oscilloscopes and multimeters. LabVIEW contains a comprehensive set of tools for acquiring, analyzing, displaying, and storing data, as well as tools to help you troubleshoot your code.

In LabVIEW, you build a user interface, or front panel, with controls and indicators. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. After you build the user interface, you add code using VIs and structures to control the front panel objects. The block diagram contains this code.

Use LabVIEW to communicate with hardware such as data acquisition, vision, and motion control devices and GPIB, PXI, VXI, RS-232, and RS-485 instruments.

Building a Virtual Instrument

In the following exercises, you will build a VI that generates a signal and displays that signal in a graph. When you complete the exercises, the front panel of the VI will look similar to the front panel in Figure 1-1.

You can complete the exercises in approximately 40 minutes.

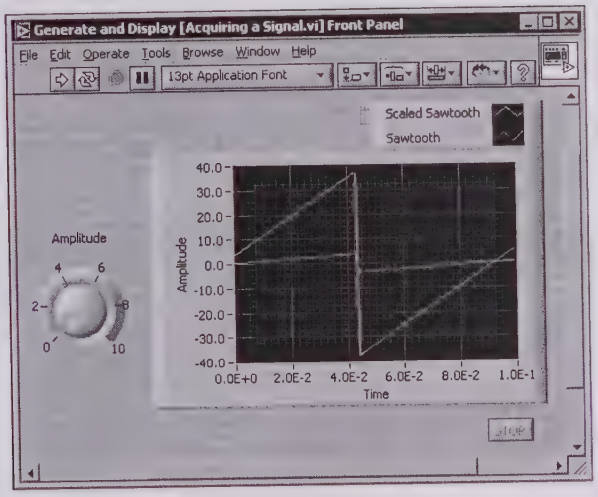


Figure 1-1. Front Panel for the Acquiring a Signal VI

Opening a New VI from a Template

LabVIEW provides templates containing information from which you can build your VI. These templates help you get started with LabVIEW. Complete the following steps to create a VI that generates a signal and displays it on the front panel.

1. Launch LabVIEW.
2. In the **LabVIEW** dialog box, shown in Figure 1-2, click the **New** button to display the **New** dialog box.

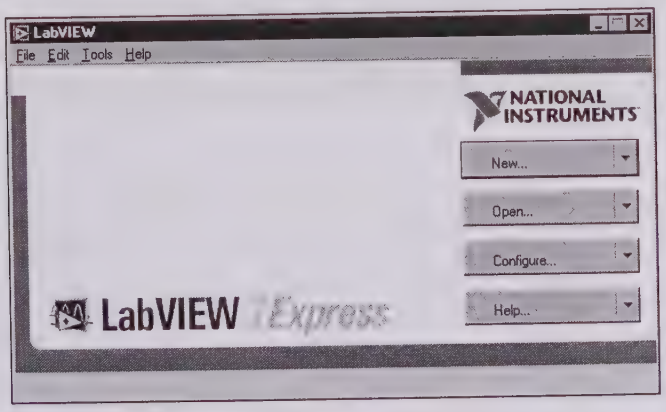


Figure 1-2. LabVIEW Dialog Box

3. Select **VI from Template»Tutorial (Getting Started)»Generate and Display** in the **Create new** list. This template VI generates and displays a signal.

   Notice that previews of the template VI appear in the **Front panel preview** and the **Block diagram preview** sections. Figure 1-3 shows the **New** dialog box and the Generate and Display template VI.

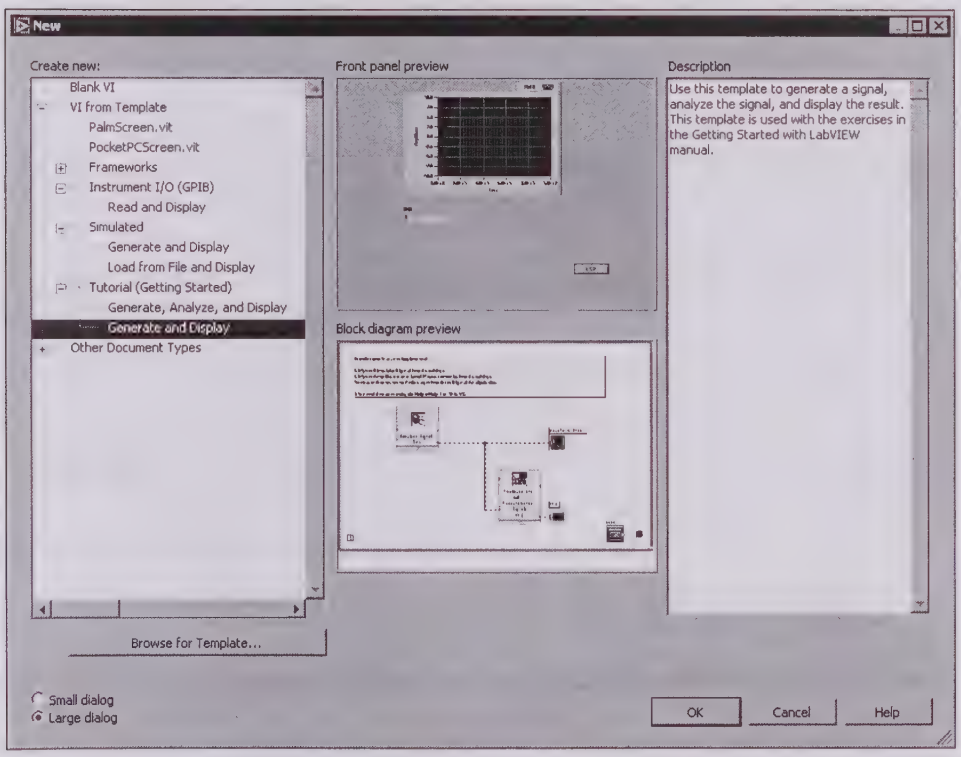


Figure 1-3. New Dialog Box

4. Click the **OK** button to open the template. You also can double-click the name of the template VI in the **Create new** list to open the template.

5. Examine the front panel of the VI.

   The user interface, or front panel, appears with a gray background and includes controls and indicators. The title bar of the front panel indicates that this window is the front panel for the Generate and Display VI.

Note If the front panel is not visible, you can display the front panel by selecting **Window»Show Front Panel**.

6. Examine the block diagram of the VI.

   The block diagram appears with a white background and includes VIs and structures that control the front panel objects. The title bar of the block diagram indicates that this window is the block diagram for the Generate and Display VI.

Note If the block diagram is not visible, you can display the block diagram by selecting **Window»Show Block Diagram**.

7. In the front panel toolbar, click the **Run** button, shown at left.

   Notice that a sine wave appears on the graph.

8. Stop the VI by clicking the **STOP** button, shown at left, on the front panel.

Adding a Control to the Front Panel

Controls on the front panel simulate the input devices on a physical instrument and supply data to the block diagram of the VI. Many physical instruments have knobs you can turn to change an input value. Complete the following steps to add a knob control to the front panel.

Tip Throughout these exercises, you can undo recent edits by selecting **Edit»Undo** or pressing the <Ctrl-Z> keys.

1. If the **Controls** palette, shown in Figure 1-4, is not visible on the front panel, select **Window»Show Controls Palette** to display it.

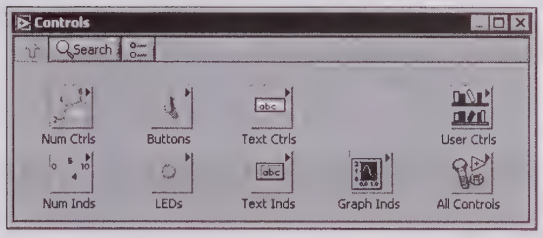


Figure 1-4. Controls Palette

2. Move the cursor over the icons on the **Controls** palette to locate the **Numeric Controls** palette.

   Notice that when you move the cursor over icons on the **Controls** palette, the name of that subpalette appears in the gray space above all the icons on the palette. When you idle the cursor over any icon on any palette, the full name of the subpalette, control, or indicator appears.

3. Click the **Numeric Controls** icon to access the **Numeric Controls** palette.

4. Select the knob control on the **Numeric Controls** palette and place it on the front panel to the left of the waveform graph.

   You will use this knob in a later exercise to control the amplitude of a signal.

5. Select **File»Save As** and save this VI as `Acquiring a Signal.vi` to an easily accessible location.

Changing the Signal Type

The block diagram has a blue icon labeled **Simulate Signal**. This icon represents the Simulate Signal Express VI. The Simulate Signal Express VI simulates a sine wave by default. Complete the following steps to change this signal to a sawtooth wave.

1. Display the block diagram by selecting **Window»Show Block Diagram** or by clicking the block diagram.

   Notice the Simulate Signal Express VI, shown at left. An Express VI is a component of the block diagram that you can configure to perform common measurement tasks. The Simulate Signal Express VI simulates a signal based on the configuration that you specify.

2. Right-click the Simulate Signal Express VI and select **Properties** from the shortcut menu to display the **Configure Simulate Signal** dialog box.

3. Select **Sawtooth** from the **Signal type** pull-down menu.

   Notice that the waveform on the graph in the **Result Preview** section changes to a sawtooth wave. The **Configure Simulate Signal** dialog box should appear similar to Figure 1-5.

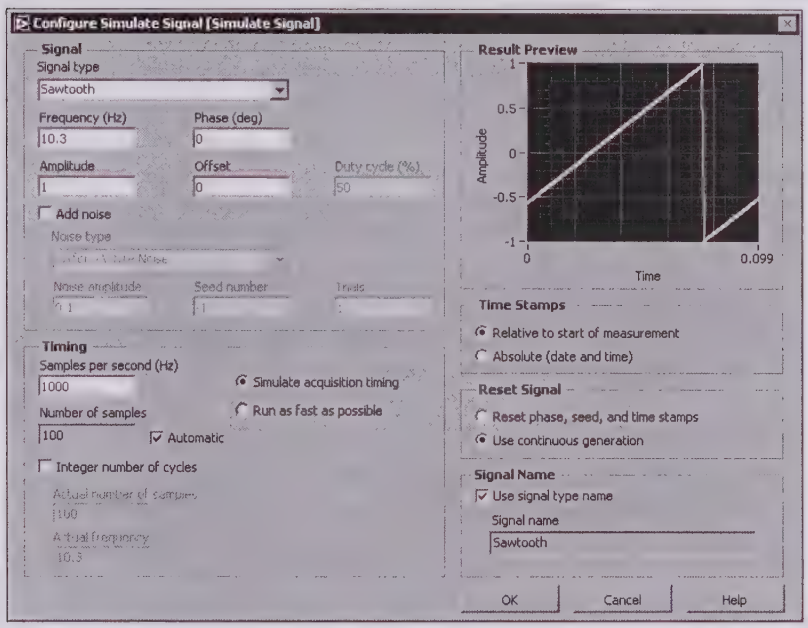


Figure 1-5. Configure Simulate Signal Dialog Box

4. Click the **OK** button to apply the current configuration and close the **Configure Simulate Signal** dialog box.
5. Move the cursor over the down arrows at the bottom of the Simulate Signal Express VI.

6. When a double-headed arrow appears, shown at left, click and drag the border of the Express VI until the **Amplitude** input appears.

   Notice how you expanded the Simulate Signal Express VI to display a new input. Because the **Amplitude** input appears on the block diagram, you can configure the amplitude of the sawtooth wave on the block diagram.

   In Figure 1-5, notice how **Amplitude** is an option in the **Configure Simulate Signal** dialog box. When inputs, such as **Amplitude**, appear on the block diagram and in the configuration dialog box, you can configure the inputs in either location.

Wiring Objects on the Block Diagram

To use the knob control to change the amplitude of the signal, you must connect the two objects on the block diagram. Complete the following steps to wire the knob to the **Amplitude** input on the Simulate Signal Express VI.

1. Move the cursor over the **Knob** terminal, shown at left, until the Positioning tool appears.

   Notice how the cursor becomes an arrow, or the Positioning tool, shown at left. Use the Positioning tool to select, position, and resize objects.

2. Click the **Knob** terminal to select it, then drag the terminal to the left of the Simulate Signal Express VI. Make sure the **Knob** terminal is inside the loop, shown at left.

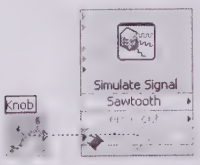
   The terminals are representations of front panel controls and indicators. Terminals are entry and exit ports that exchange information between the front panel and block diagram.

3. Deselect the **Knob** terminal by clicking a blank space on the block diagram.

4. Move the cursor over the arrow of the **Knob** terminal, shown at left.

   Notice how the cursor becomes a wire spool, or the Wiring tool, shown at left. Use the Wiring tool to wire objects together on the block diagram.

Note The cursor does not switch to another tool while an object is selected.



5. When the Wiring tool appears, click the arrow and then click the **Amplitude** input of the Simulate Signal Express VI, shown at left, to wire the two objects together.

   Notice that a wire appears and connects the two objects. Data flows along this wire from the terminal to the Express VI.

6. Select **File»Save** to save this VI.

Running the VI

Running a VI executes your solution. Complete the following steps to run the Acquiring a Signal VI.

1. Display the front panel by selecting **Window»Show Front Panel** or by clicking the front panel.

Tip Press the <Ctrl-E> keys to switch from the front panel to the block diagram or from the block diagram to the front panel.

2. Click the **Run** button.
3. Move the cursor over the knob control.

Notice how the cursor becomes a hand, or the Operating tool, shown at left. Use the Operating tool to change the value of a control or select the text within a control.

4. Using the Operating tool, turn the knob to adjust the amplitude of the sawtooth wave.

Notice how the amplitude of the sawtooth wave changes as you turn the knob. Also notice that the y-axis on the graph autoscales to account for the change in amplitude.

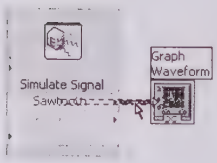
To indicate that the VI is running, the **Run** button changes to a darkened arrow, shown at left. You cannot edit the front panel or block diagram while the VI runs.

5. Click the **STOP** button, shown at left, to stop the VI.

Note Although the **Abort Execution** button looks like a stop button, the **Abort Execution** button does not always properly close the VI. National Instruments recommends stopping your VIs using the **STOP** button on the front panel. Use the **Abort Execution** button only when errors prevent you from terminating the application using the **STOP** button.

Modifying the Signal

Complete the following steps to add scaling to the signal and display the results in the graph on the front panel.



1. On the block diagram, use the Positioning tool to double-click the wire that connects the Simulate Signal Express VI to the **Waveform Graph** terminal, shown at left.
2. Press the <Delete> key to delete this wire.
3. If the **Functions** palette, shown in Figure 1-6, is not visible, select **Window»Show Functions Palette** to display it.

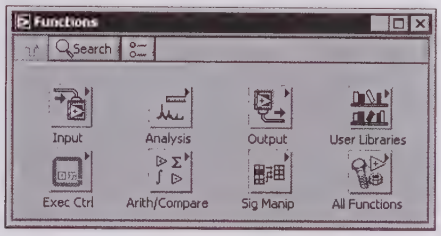


Figure 1-6. Functions Palette

4. Select the Scaling and Mapping Express VI, shown at left, on the **Arithmetic & Comparison** palette and place it on the block diagram inside the loop between the Simulate Signal Express VI and the **Waveform Graph** terminal. If there is no room between the Express VI and the terminal, move the **Waveform Graph** terminal to the right.

   Notice that the **Configure Scaling and Mapping** dialog box automatically opens when you place the Express VI on the block diagram.

5. Define the value of the scaling factor by entering `10` in the **Slope (m)** text box.

   The **Configure Scaling and Mapping** dialog box should appear similar to Figure 1-7.

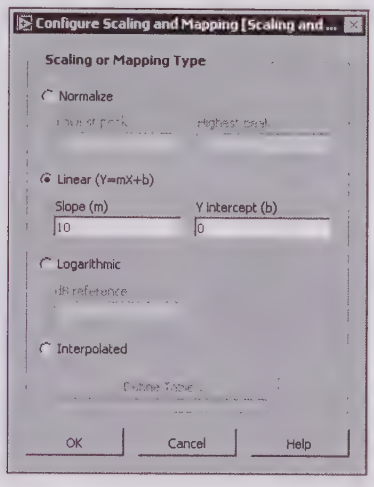
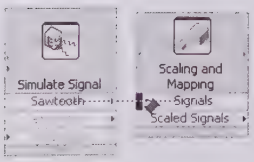


Figure 1-7. Configure Scaling and Mapping Dialog Box

6. Click the **OK** button to apply the current configuration and close the **Configure Scaling and Mapping** dialog box.
7. Move the cursor over the arrow on the **Sawtooth** output of the Simulate Signal Express VI.
8. When the Wiring tool appears, click the arrow and then click the arrow on the **Signals** input of the Scaling and Mapping Express VI, shown at left, to wire the two objects together.



9. Using the Wiring tool, wire the **Scaled Signals** output of the Scaling and Mapping Express VI to the **Waveform Graph** terminal.

   Notice the wires connecting the Express VIs and terminals. The arrows on the Express VIs and terminals indicate the direction that the data flows along these wires. The block diagram should appear similar to Figure 1-8.

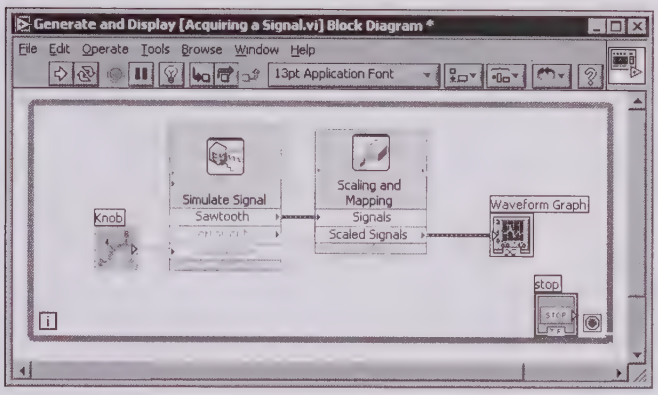


Figure 1-8. Block Diagram for the Acquiring a Signal VI

10. Select **File»Save** to save this VI.

Displaying Two Signals on the Graph

To compare the signal generated by the Simulate Signal Express VI and the signal modified by the Scaling and Mapping Express VI on the same graph, use the Merge Signals function. Complete the following steps to display two signals on the same graph.

1. Move the cursor over the arrow on the **Sawtooth** output of the Simulate Signal Express VI.

2. Using the Wiring tool, wire the **Sawtooth** output to the **Waveform Graph** terminal.

The Merge Signals function, shown at left, appears where the two wires connect. This function takes the two separate signals and combines them so that both can be displayed on the same graph. The block diagram should appear similar to Figure 1-9.

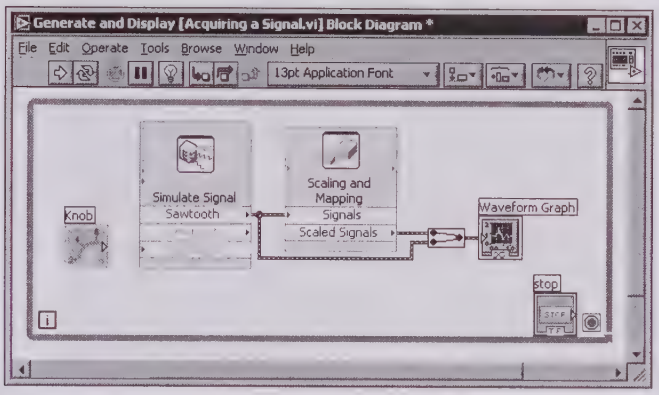


Figure 1-9. Block Diagram Showing the Merge Signals Function

3. Select **File»Save** to save this VI. You also can press the <Ctrl-S> keys to save a VI.
4. Return to the front panel, run the VI, and turn the knob control.

   Notice that the graph plots the sawtooth wave and the scaled signal. Also notice that the maximum value on the y-axis automatically changes to be 10 times the knob value. This scaling occurs because you set the slope to 10 in the Scaling and Mapping Express VI.
5. Click the **STOP** button.

Customizing the Knob

The knob control changes the amplitude of the sawtooth wave so labeling it **Amplitude** accurately describes the function of the knob. Complete the following steps to customize the appearance of a control on the front panel.

1. Right-click the knob and select **Properties** from the shortcut menu to display the **Knob Properties** dialog box.
2. In the **Label** section on the **Appearance** tab, delete the label **Knob**, and enter `Amplitude` in the text box.

The **Knob Properties** dialog box should appear similar to Figure 1-10.

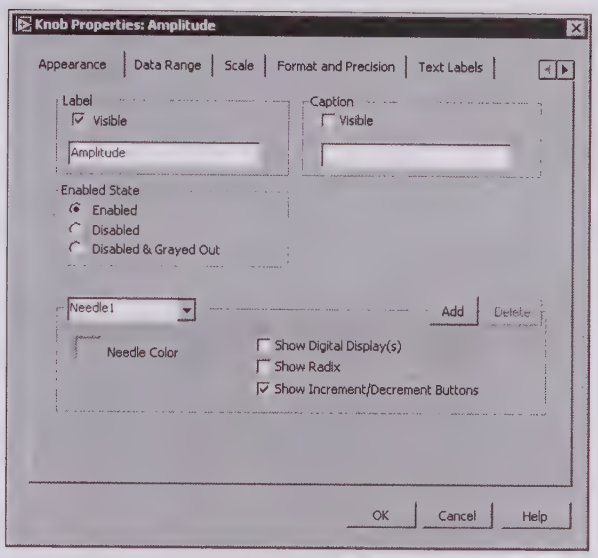


Figure 1-10. Knob Properties Dialog Box

3. Click the **Scale** tab and in the **Scale Style** section, place a checkmark in the **Ramp Visible** checkbox.

   Notice how the knob on the front panel instantly updates to reflect these changes.

4. Click the **OK** button to apply the current configuration and close the **Knob Properties** dialog box.
5. Save this VI.

Tip As you build your VIs, you can experiment with different properties and configurations. You also can add and delete objects. Remember you can undo recent edits by selecting **Edit»Undo** or pressing the <Ctrl-Z> keys.

6. Experiment with other properties of the knob by using the **Knob Properties** dialog box. For example, try changing the colors for the **Marker Text Color** by clicking the color box located on the **Scale** tab.
7. Click the **Cancel** button to avoid applying the changes you made while experimenting. If you want to keep the changes you made, click the **OK** button.

Customizing the Waveform Graph

The waveform graph indicator displays the two signals. To indicate which plot is the scaled signal and which is the simulated signal, you customize the plots. Complete the following steps to customize the appearance of an indicator on the front panel.

1. Move the cursor over the top of the plot legend on the waveform graph.

   Notice that while there are two plots on the graph, the plot legend displays only one plot.

2. When a double-headed arrow appears, shown in Figure 1-11, click and drag the border of the plot legend until the second plot name appears.

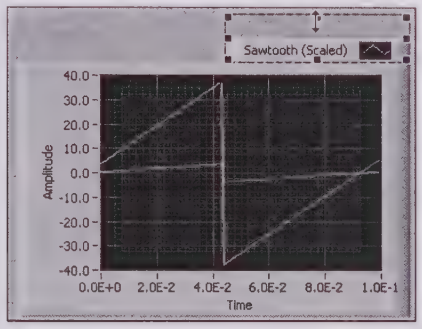


Figure 1-11. Expanding a Plot Legend

3. Right-click the waveform graph and select **Properties** from the shortcut menu to display the **Graph Properties** dialog box.
4. On the **Plots** tab, select **Sawtooth** from the pull-down menu. Click the **Line Color** color box to display the color picker. Select a new line color.
5. Select **Sawtooth (Scaled)** from the pull-down menu.
6. Place a checkmark in the **Don't use waveform names for plot names** checkbox.
7. In the **Name** text box, delete the current label and change the name of this plot to `Scaled Sawtooth`.

8. Click the **OK** button to apply the current configuration and close the **Graph Properties** dialog box.

   Notice how the plot color on the front panel changes.

9. Experiment with other properties of the graph by using the **Graph Properties** dialog box. For example, try disabling the autoscale feature located on the **Scales** tab.

10. Click the **Cancel** button to avoid applying the changes you made while experimenting. If you want to keep the changes you made, click the **OK** button.

11. Save and close this VI.

Summary

The following topics are a summary of the main concepts you learned in this chapter.

New Dialog Box and VI Templates

The **New** dialog box contains many LabVIEW VI templates, including the ones used in this manual. The VI templates help you start building VIs for common measurements and other tasks. The VI templates include the Express VIs, functions, and front panel objects you need to get started building common measurement applications.

You can use one of the following methods to access the **New** dialog box.

- Click the **New** button on the **LabVIEW** dialog box.
- Click the arrow on the **New** button in the LabVIEW dialog box and select **New** from the **New** pull-down menu.
- Select **File»New** from the front panel or block diagram menu bar.

Front Panel

The front panel is the user interface of a VI. You build the front panel by using controls and indicators, which are the interactive input and output terminals of the VI, respectively. Controls and indicators are located on the **Controls** palette.

Controls are knobs, push buttons, dials, and other input devices. Controls simulate the input devices on a physical instrument and supply data to the block diagram of the VI.

Indicators are graphs, LEDs, and other displays. Indicators simulate the output devices on a physical instrument and display data the block diagram acquires or generates.

Property Dialog Boxes

Use property dialog boxes or shortcut menus to configure how controls and indicators appear or behave on the front panel. Right-click a control or indicator on the front panel and select **Properties** from the shortcut menu to access the property dialog box for that object. You cannot access property dialog boxes for a control or indicator when the VI is running.

Block Diagram

The block diagram contains the graphical source code for how the VI runs. Front panel objects appear as icon terminals on the block diagram. Wires connect control and indicator terminals to Express VIs. Data flows from controls to Express VIs, from Express VIs to Express VIs, and from Express VIs to indicators through the wires.

Express VIs

Use Express VIs located on the **Functions** palette for common measurement tasks. When you place an Express VI on the block diagram, the dialog box you use to configure that Express VI appears by default. Set the options in this configuration dialog box to specify how an Express VI behaves.

Express VIs appear on the block diagram as expandable nodes with icons surrounded by a blue field. You can resize an Express VI to display its inputs and outputs. Inputs accept data, and outputs return data. The inputs and outputs for the Express VI depend on how you configure the VI.

2

Analyzing and Saving a Signal

LabVIEW includes a set of Express VIs to help you analyze signals. This chapter teaches you how to use LabVIEW to perform a basic analysis of a signal and how to save the analyzed data to a file.

Building a VI from a Template

In the following exercises, you will build a VI that generates a signal, extracts the DC value of the signal, indicates if the signal exceeds a certain limit, and records the data. When you complete the exercises, the front panel of the VI will look similar to the front panel in Figure 2-1.

You can complete the exercises in approximately 40 minutes.

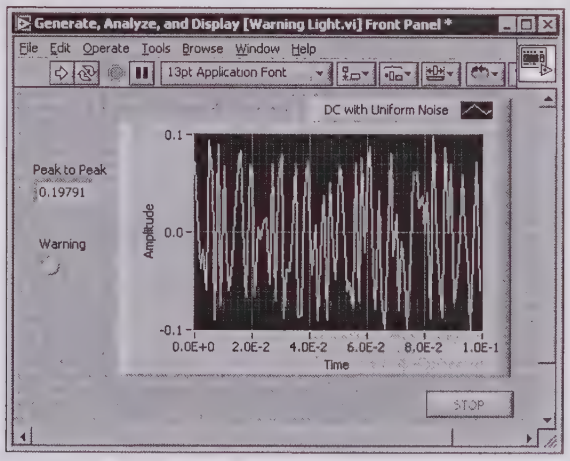


Figure 2-1. Front Panel for the Warning Light VI

Opening a New VI from a Template

To build this VI, you can start from the **New** dialog box. Complete the following steps to select a new template VI that generates, analyzes, and displays a signal.

1. In the **LabVIEW** dialog box, click the **New** button to display the **New** dialog box.

Note You also can access the **New** dialog box by clicking the arrow on the **New** button and selecting **New** from the **New** pull-down menu or by selecting **File»New** from the front panel or block diagram menu bar.

2. Select the **VI from Template»Tutorial (Getting Started)»Generate, Analyze, and Display** template in the **Create new** list.

   The template VI simulates a signal and analyzes it for its root mean square (RMS) value.

3. Click the **OK** button to open the template. You also can double-click the name of the template VI in the **Create new** list to open the template.
4. Display the block diagram by pressing the <Ctrl-E> keys.
5. If the **Context Help** window, shown in Figure 2-2, is not visible, select **Help»Show Context Help** from the block diagram menu bar to display the **Context Help** window.

Note You also can press the <Ctrl-H> keys to display the **Context Help** window.

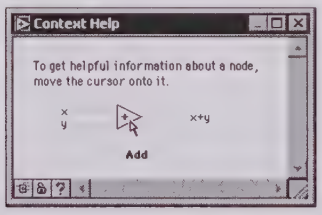
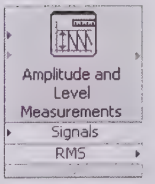


Figure 2-2. Context Help Window



6. Move the cursor over the Amplitude and Level Measurements Express VI, shown at left.

   Notice when you move the cursor over the Express VI, the **Context Help** window displays information about the Express VI, including information about how it is configured.

Keep the **Context Help** window open and notice how it provides useful information as you complete the rest of this exercise.

Modifying the Block Diagram

The Simulate Signal Express VI simulates a sine wave by default. You can customize the simulated signal by changing the options in the **Configure Simulate Signal** dialog box. Complete the following steps to change the simulated signal from a sine wave to a DC signal with uniform white noise.

1. Right-click the Simulate Signal Express VI and select **Properties** from the shortcut menu to display the **Configure Simulate Signal** dialog box.
2. Select **DC** from the **Signal type** pull-down menu.
3. Place a checkmark in the **Add noise** checkbox to add noise to the DC signal.
4. Type `0.1` in the **Noise amplitude** text box.

   Notice that the **Result Preview** section displays a random signal. The **Configure Simulate Signal** dialog box should appear similar to Figure 2-3.

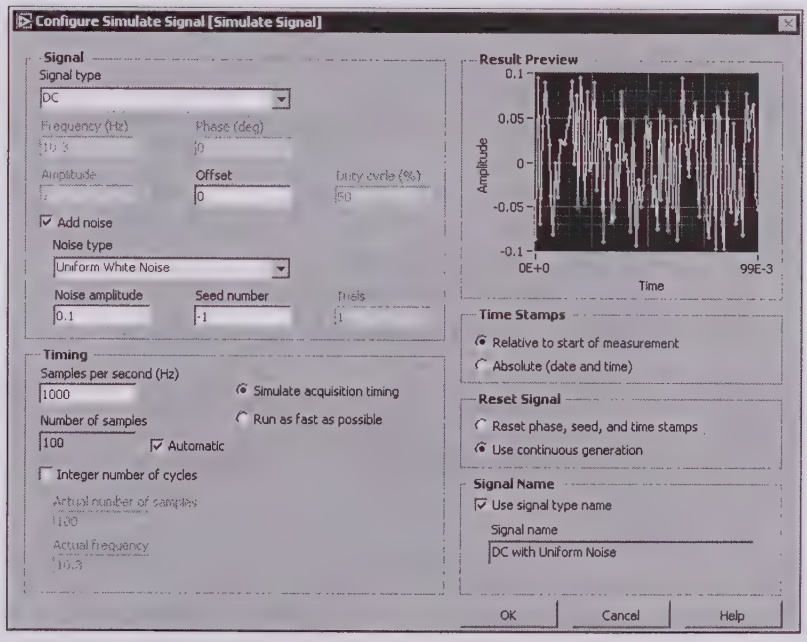


Figure 2-3. Configure Simulate Signal Dialog Box

5. Click the **OK** button to save the current configuration and close the **Configure Simulate Signal** dialog box.
6. Display the front panel by pressing the <Ctrl-E> keys.
7. Run the VI.

   Notice that the signal appears in the graph and the RMS value for the signal appears in the digital indicator.
8. Click the **STOP** button.
9. Select **File»Save As** and save this VI as `Analysis.vi` to an easily accessible location.

Modifying the Front Panel

If you decide that you do not want to use an indicator that comes with the template, you can delete it. Complete the following steps to remove the **RMS** indicator from the front panel.

1. Move the cursor over the **RMS** indicator until the Positioning tool appears.

2. Click the **RMS** indicator, shown at left, to select it and press the <Delete> key.
3. Display the block diagram.

   The block diagram now has a dashed wire with a red X, shown at left. This is a broken wire. Also notice the **Run** button, shown at left, appears broken to indicate the VI cannot run.
4. Click the broken **Run** button to display the **Error list** window.

   The **Error list** window lists all errors in the VI and provides details about each error.
5. In the **Errors and Warnings** listbox, double-click the **`Wire: has loose ends`** error to highlight the broken wire.

   Notice how LabVIEW automatically displays the problem causing the error.
6. Press the <Delete> key to delete the broken wire.

Tip Press the <Ctrl-B> keys to delete *all* broken wires on the block diagram.

7. Select **Windows»Show Error List** to display the **Error list** window. Now there are no errors listed in the **Errors and Warnings** listbox.

Note You also can press the <Ctrl-L> keys to display the **Error list** window.

8. Click the **Close** button to close this window.

   Notice that the **Run** button is no longer broken.

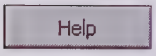
Analyzing the Amplitude of a Signal

The Amplitude and Level Measurements Express VI includes options that you can use to analyze the voltage characteristics of a signal. Complete the following steps to reconfigure the Express VI to measure the peak to peak amplitude values of the signal.

1. Right-click the Amplitude and Level Measurements Express VI and select **Properties** from the shortcut menu to display the **Configure Amplitude and Level Measurements** dialog box.

Tip You also can double-click the Express VI to open the **Configure Amplitude and Level Measurements** dialog box.

2. Remove the checkmark in the **RMS** checkbox in the **Amplitude Measurements** section.



3. Click the **Help** button, shown at left, in the bottom right corner of the **Configure Amplitude and Level Measurements** dialog box to display the *LabVIEW Help* topic for this Express VI.

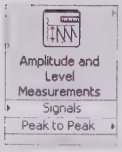
   The help topic describes the Express VI, the inputs and outputs for the Express VI, and the configuration options. Each Express VI has a corresponding help topic accessible by pressing the **Help** button.

4. In the *Amplitude and Level Measurements* topic, find the output parameter whose description indicates that it takes a measurement from the maximum peak to the minimum peak of the signal.

5. Minimize *LabVIEW Help* to return to the **Configure Amplitude and Level Measurements** dialog box.

6. Select the input or output you decided to use.

   Notice how the option you selected, **Peak to Peak**, appears in the **Results** section with the corresponding value of the measurement.



7. Click the **OK** button to close the **Configure Amplitude and Level Measurements** dialog box and return to the block diagram.

   Notice that the RMS output in the Amplitude and Level Measurements Express VI changed to reflect the new **Peak to Peak** parameter, shown at left.

Adding a Warning Light

If you want a visual cue indicating when a value exceeds a specified limit, use a warning light. Complete the following steps to add a warning light to the VI.

1. From the **Controls** palette, select the round LED indicator on the **LEDs** palette, shown in Figure 2-4, and place it on the front panel to the left of the waveform graph.

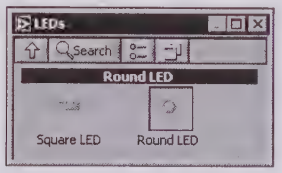


Figure 2-4. LEDs Palette

2. Right-click the LED and select **Properties** from the shortcut menu to display the **Button Properties** dialog box.
3. Change the label of the LED to `Warning`.
4. Click the **OK** button to save the current configuration and close the **Button Properties** dialog box.

   You will use this LED in a later exercise to signal when a value has exceeded its limit.

5. Select **File»Save As** and save this VI as `Warning Light.vi` to an easily accessible location.

Setting the Warning Level Limit

To specify the value at which you want the warning light to turn on, use the Comparison Express VI. Complete the following steps to compare the peak to peak value to a limit you set.

1. On the block diagram, select the Comparison Express VI on the **Arithmetic & Comparison»Express Comparison** palette and place it to the right of the Amplitude and Level Measurements Express VI.
2. In the **Configure Comparison** dialog box, select the **> Greater than** option from the **Compare Condition** section.
3. In the **Comparison Inputs** section, select **Use constant value** and type `0.195` in the **Constant value** text box.
4. Close the configuration page and return to the block diagram.

   Notice how the name of the Comparison Express VI reflects the operation of the Express VI, shown at left. **Greater Than** indicates that the Express VI does a greater than comparison.
5. Wire the **Peak to Peak** output of the Amplitude and Level Measurements Express VI to the **Operand 1** input of the Comparison Express VI.
6. Move the cursor over the wire connecting the **Peak to Peak** output to the **Operand 1** input.
7. When the Positioning tool appears, right-click the wire connecting the **Peak to Peak** output to the **Operand 1** input and select **Create» Numeric Indicator** from the shortcut menu.

   Notice how a **Peak to Peak** terminal, shown at left, appears on the block diagram. If the **Peak to Peak** terminal appears to be on top of the wires of the Express VIs, move the Express VIs and **Peak to Peak** terminal around to create more space. For example, move the **Peak to Peak** terminal into the blank space above the Express VIs.

Warning the User

After specifying the values at which you want the warning light to turn on, you must wire the warning light to the Comparison Express VI. Complete the following steps to provide a visual cue to the user when the peak to peak value of the signal exceeds a specified limit.

1. On the block diagram, move the **Warning** terminal to the right of the Comparison Express VI. Make sure the **Warning** terminal is inside the loop, as shown in Figure 2-5.
2. Wire the **Result** output of the Comparison Express VI to the **Warning** terminal.

   The block diagram should appear similar to Figure 2-5.

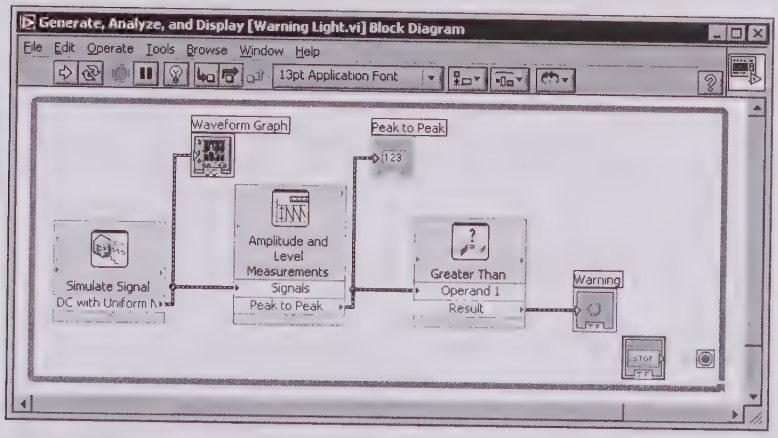


Figure 2-5. Block Diagram for the Warning Light VI

3. Display the front panel.

   Notice how a numeric indicator also appears on the front panel labeled **Peak to Peak**. This indicator displays the peak to peak value of the signal.
4. Run the VI.

   Notice that when the peak to peak value exceeds 0.195, the **Warning** indicator lights up.
5. Click the **STOP** button to stop the VI.
6. Select **File»Save** to save this VI.

Configuring the VI to Save Data to a File

To store information about the data your VI generates, use the Write LabVIEW Measurement File Express VI. Complete the following steps to build a VI that saves peak to peak values and other information to a LabVIEW data file.

1. Select the Write LabVIEW Measurement File Express VI on the **Output** palette and place it on the block diagram below and to the right of the Amplitude and Level Measurements Express VI.

   Notice that the **File name** text box indicates that the output file is `test.lvm` and displays the full path to the `test.lvm` file. A `.lvm` file is a LabVIEW measurement data file, which LabVIEW places in the default `LabVIEW Data` directory. LabVIEW installs the `LabVIEW Data` directory in the default file directory of the operating system.

   When you want to view the data, use the file path displayed in the **File name** text box to access the `test.lvm` file.

2. In the **Configure Write LabVIEW Measurement File** dialog box, select the **Append to file** option in the **If a file already exists** section.

   By selecting **Append to file**, LabVIEW writes all the data to the `test.lvm` file without erasing the existing data in the file if a file by that name exists already.

3. Select the **One header only** option in the **Segment Headers** section.

4. Enter the following text in the **File Description** text box: `Sample of peak to peak values.`

5. Close the **Configure Write LabVIEW Measurement File** dialog box and return to the block diagram.

Saving Data to a File

When you run this VI, LabVIEW saves the data to the `test.lvm` file. Complete the following steps to generate the `test.lvm` file.

1. Wire the **Peak to Peak** output of the Amplitude and Level Measurements Express VI to the **Signals** input of the Write LabVIEW Measurement File Express VI.

2. Select **File»Save As** and save this VI as `Save Data.vi` to an easily accessible location.

3. Display the front panel and run the VI.

4. Click the **STOP** button on the front panel.

5. To view the data you saved, open the `LabVIEW Data\test.lvm` file with a spreadsheet or word processing application.
6. Close the file when you finish looking at it and return to the Save Data VI.

Adding a Button that Stores Data when Pressed

If you want to store only certain data, you can configure the Write LabVIEW Measurement File Express VI to save peak to peak values only when a user presses a button. Complete the following steps to add a button to the VI and configure how the button responds when a user clicks it.

1. On the front panel, select the rocker button on the **Buttons & Switches** palette and place it to the right of the waveform graph.
2. Using the **Button Properties** dialog box, change the label of the button to `Write to File`.
3. On the **Operation** tab, select **Latched When Pressed** from the **Button Behavior** list.

   Use the **Operation** tab to specify how a button behaves when a user clicks it. To see how the button reacts to a click, click the button in the **Preview Selected Behavior** section.
4. Close the **Button Properties** dialog box.
5. Save this VI.

Saving Data when Prompted by a User

Complete the following steps to build a VI that logs data to a file when the user clicks a button on the front panel.

1. On the block diagram, double-click the Write LabVIEW Measurement File Express VI to access the **Configure Write LabVIEW Measurement File** dialog box.
2. Change the file name `test.lvm` to `Selected Samples.lvm` in the **File name** text box to save the data to a different file.
3. Close the **Configure Write LabVIEW Measurement File** dialog box.
4. Right-click the **Signal** input of the Write LabVIEW Measurement File Express VI. Select **Insert Input/Output** from the shortcut menu to insert the **Comment** input.

5. Right-click the **Comment** input of the Write LabVIEW Measurement File Express VI. Select **Select Input/Output»Enable** from the shortcut menu to insert the **Enable** input.

   In the previous exercise you learned to add inputs and outputs by expanding the Express VI using the down arrows. Notice that this method is a different way of displaying and selecting the inputs and outputs of an Express VI.

   The inputs and outputs of an Express VI appear in a predetermined order when you add new inputs and outputs. To select a specific input, you may need to add an input first, then change the input to the specific one you want to use.

6. Move the **Write to File** terminal to the left of the Write LabVIEW Measurement File Express VI.

7. Wire the **Write to File** terminal to the **Enable** input of the Write LabVIEW Measurement File Express VI.

   The block diagram should appear similar to Figure 2-6.

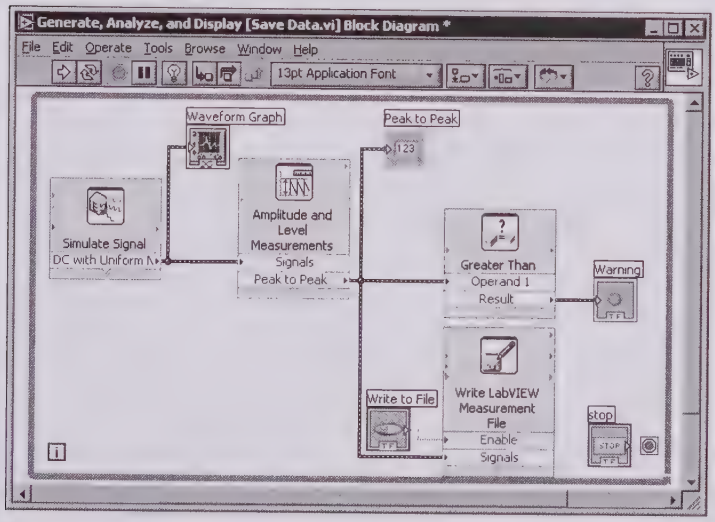


Figure 2-6. Block Diagram for the Save Data VI

8. Display the front panel and run the VI. Click the **Write to File** button several times.

9. Click the **STOP** button on the front panel.

10. To view the data you saved, open the `Selected Samples.lvm` file with a spreadsheet or word processing application.

    Notice how the `Selected Samples.lvm` file differs from the `test.lvm` file. `test.lvm` recorded all the data generated by the Save Data VI, whereas `Selected Samples.lvm` only recorded the data when you pressed the **Write to File** button.

11. Save and close this VI.

Summary

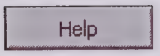
The following topics are a summary of the main concepts you learned in this chapter.

LabVIEW Help Resources

LabVIEW includes extensive documentation for new and experienced LabVIEW users. Help resources include manuals, the Context Help window, *LabVIEW Help*, examples, and Application Notes.

Note All LabVIEW manuals and Application Notes also are available as PDFs. You must have Adobe Acrobat Reader with Search and Accessibility 5.0.5 or later to access the PDFs. Refer to the Adobe Systems Incorporated Web site at `www.adobe.com` to download Acrobat Reader.

The **Context Help** window displays basic information about LabVIEW objects when you move the cursor over each object. To access the **Context Help** window, select **Help»Show Context Help**. You also can press the <Ctrl-H> keys to display the **Context Help** window.



The *LabVIEW Help* contains reference information about LabVIEW Express VIs, controls, indicators, palettes, menus, tools, VIs, and functions. This help file also includes step-by-step instructions for using LabVIEW features. To access help information for Express VIs, click the **Help** button, shown at left, in the configuration dialog box while configuring an Express VI. You also can access the *LabVIEW Help* by selecting **Help»VI, Function, & How-To Help** or pressing the <Ctrl-?> keys.

Controls and Indicators

You can configure the controls and indicators on the front panel to perform tasks depending on what you want your VI to do. The following is a summary of the different ways you learned to use controls and indicators in this chapter.

- You can build VIs that automatically perform a task whenever certain conditions are met, such as displaying a warning light when a value exceeds a certain limit.
- You can build VIs that let users control when an Express VI executes using buttons and the **Enable** input. You can configure the buttons to operate in one of six ways using the **Operations** tab of the **Button Properties** dialog box.

Saving Data

The Write LabVIEW Measurement File Express VI saves data that a VI generates and analyzes to a data file. The LabVIEW measurement data file (`.lvm`) is a tab-delimited text file you can open with a spreadsheet application or a text-editing application. In addition to the data an Express VI generates, the `.lvm` file includes headers that contain information about the data, such as the date and time the data was generated.

To store the data files LabVIEW generates, LabVIEW installs the `LabVIEW Data` directory in the default file directory of the operating system to help you organize and locate the data files LabVIEW generates.

Note Refer to the *LabVIEW Help* for information about saving data to and retrieving data from a `.lvm` file.

Errors and Broken Wires

The **Run** button appears broken when the VI you are creating or editing contains errors. If the **Run** button is still broken when you finish wiring the block diagram, the VI is broken and cannot run.

Click the broken **Run** button or select **Window»Show Error List** to find out why a VI is broken. You can use the **Error list** window to locate errors on the block diagram. Double-click the error in the **Errors and Warnings** listbox to highlight the problem causing the error.

A broken wire appears as a dashed black line with a red X in the middle. Broken wires occur for a variety of reasons, such as deleting wired objects. The VI cannot run if the block diagram includes broken wires.

Move the Wiring tool over a broken wire to display a tip strip that describes why the wire is broken. This information also appears in the **Context Help** window when you move the Wiring tool over a broken wire. Right-click the wire and select **List Errors** from the shortcut menu to display the **Error list** window. Click the **Help** button for more information about why the wire is broken.

3

Expanding the Features of a VI

You can choose one of many LabVIEW VI templates to use as a starting point. However, sometimes you need to build a VI for which a template is not available. This chapter teaches you how to build and customize a VI without using a template.

Building a VI from a Blank Template

In the following exercises, you will open a blank VI and add Express VIs and structures to the block diagram to build a new VI. You will build a VI that generates a signal, reduces the number of samples in the signal, and displays the resulting data in a table on the front panel. When you complete the exercises, the front panel of the VI will look similar to the front panel in Figure 3-1.

You can complete the exercises in approximately 30 minutes.

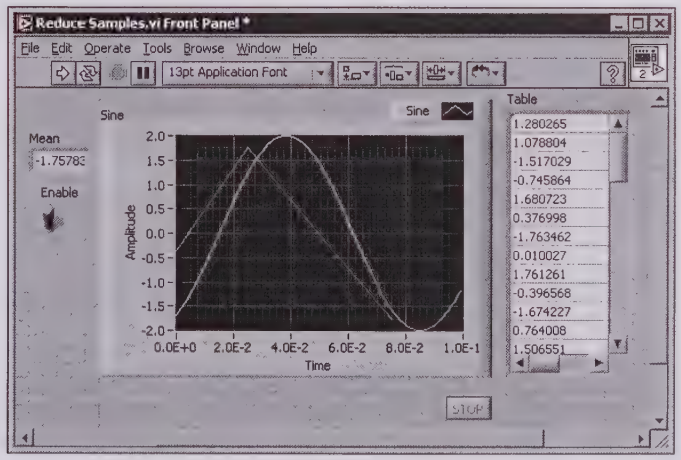


Figure 3-1. Front Panel for the Reduce Samples VI

Opening a Blank VI

If no template is available for the task you want to create, you can start with a blank VI and add Express VIs to accomplish the specific task. Complete the following steps to open a blank VI.

1. In the **LabVIEW** dialog box, click the arrow on the **New** button and select **Blank VI** from the shortcut menu or press the <Ctrl-N> keys to open a blank VI.

   Notice that a blank front panel and block diagram appear.

Note You also can open a blank VI by selecting **Blank VI** from the **Create new** list in the **New** dialog box or by selecting **File»New VI** from the front panel or block diagram menu bar.

2. If the **Functions** palette is not visible, right-click any blank space on the block diagram to bring up the temporary version of the **Functions** palette. Click the thumbtack, shown at left, in the upper left corner of the **Functions** palette to place the palette on the screen.

Note You can right-click on a blank space on the block diagram or the front panel to bring up the **Functions** or **Controls** palettes.

Adding an Express VI that Simulates a Signal

Complete the following steps to find the Express VI you want to use and then add it to the block diagram.

1. If the **Context Help** window is not visible, press the <Ctrl-H> keys to open the **Context Help** window. You also can press the **Show Context Help Window** button, shown at left, to open the **Context Help** window.
2. Select the **Input** palette on the **Functions** palette and move the cursor over the Express VIs on the **Input** palette.

   Notice that the **Context Help** window displays information about the function of each Express VI.
3. From the information provided in the **Context Help** window, find the Express VI that can output a sine wave signal.
4. Select the Express VI and place it on the block diagram. The **Configure Simulate Signal** dialog box appears.
5. Idle the cursor over the various options in the **Configure Simulate Signal** dialog box, such as **Frequency (Hz)**, **Amplitude**, and **Samples**

per second (Hz). Read the information that appears in the **Context Help** window.

6. Configure the Simulate Signal Express VI to generate a sine wave with a frequency of `10.7` and amplitude of `2`.
7. Notice how the signal displayed in the **Result Preview** window changes to reflect the configured sine wave.
8. Close the **Configure Simulate Signal** dialog box.
9. Move the cursor over the Simulate Signal Express VI and read the information that appears in the **Context Help** window.

   Notice that the **Context Help** window now displays the configuration of the Simulate Signal Express VI.
10. Save this VI as `Reduce Samples.vi` to an easily accessible location.

Modifying the Signal

Complete the following steps to use the *LabVIEW Help* to search for the Express VI that reduces the number of samples in a signal.

1. Select **Help»VI, Function, & How-To Help** to open the *LabVIEW Help*.
2. Click the **Search** tab and type `sample compression` in the **Type in the word(s) to search for** text box.

   Notice that this word choice reflects what you want this Express VI to do—compress, or reduce, the number of samples in a signal.
3. Select the *Sample Compression Express VI* topic to display the topic that describes the Sample Compression Express VI.
4. After you read the description of the Express VI, click the **Place on the block diagram** button, shown at left, to select the Express VI.
5. Move the cursor to the block diagram.

   Notice how LabVIEW attaches the Sample Compression Express VI to the cursor.
6. Place the Sample Compression Express VI on the block diagram to the right of the Simulate Signal Express VI.
7. Configure the Sample Compression Express VI to reduce the signal by a factor of 25 using the mean of these values.
8. Close the **Configure Sample Compression** dialog box.
9. Using the Wiring tool, wire the **Sine** output in the Simulate Signal Express VI to the **Signals** input in the Sample Compression Express VI.

Customizing the Front Panel

In the previous exercises, you added controls and indicators to the front panel using the **Controls** palette. You also can add controls and indicators from the block diagram. Complete the following steps to create controls and indicators.

1. Right-click the **Mean** output in the Sample Compression Express VI and select **Create»Numeric Indicator** to create a numeric indicator.
2. Right-click the **Mean** output of the Sample Compression Express VI and select **Insert Input/Output** from the shortcut menu to insert the **Enable** input.
3. Right-click the **Enable** input and select **Create»Control** to create the **Enable** switch.
4. Right-click the wire linking the **Sine** output in the Simulate Signal Express VI to the **Signals** input in the Signal Compression Express VI and select **Create»Graph Indicator**.

   Notice that you can create controls and indicators from the block diagram. When you create controls and indicators using this method, LabVIEW automatically creates terminals that are labeled and formatted correctly.
5. Using the Wiring tool, wire the **Mean** output in the Sample Compression Express VI to the **Sine** terminal.

   Notice that the Merge Signals function appears.
6. Arrange the objects on the block diagram so that they appear similar to Figure 3-2.

Tip You can right-click any wire and select **Clean Up Wire** from the shortcut menu to have LabVIEW automatically arrange the wires for you.

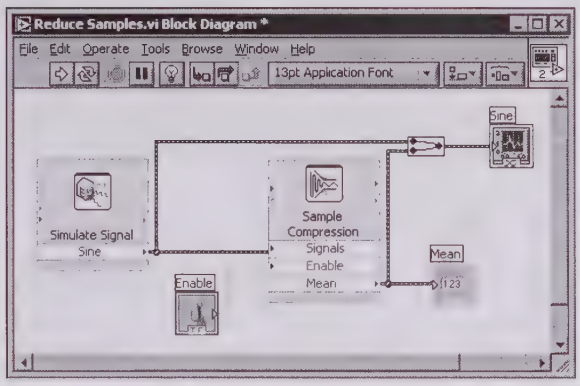


Figure 3-2. Block Diagram for the Reduce Samples VI

7. Display the front panel.

   Notice that the controls and indicators you added automatically appear on the front panel with labels that correspond to their function.

8. Save this VI.

Configuring the VI to Run Continuously Until the User Stops It

In the current state, the VI runs once, generates one signal, then stops executing. To run the VI until a condition is met, you can add a While Loop to the block diagram. Complete the following steps to add a While Loop.

1. Display the front panel and run the VI.

   Notice how the VI runs once and then stops. Also notice how there is no **STOP** button.

2. Display the block diagram and select the While Loop on the **Execution Control** palette.

3. Move the cursor to the upper left corner of the block diagram. Place the top left corner of the While Loop here.

4. Click and drag the cursor diagonally to enclose *all* the Express VIs and wires, as shown in Figure 3-3.

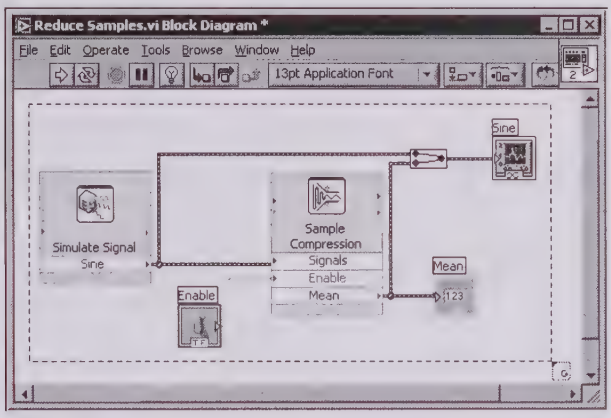


Figure 3-3. Placing the While Loop around the Express VIs

Notice that the While Loop, shown at left, appears with a **STOP** button wired to the condition terminal. This While Loop is configured to stop when the user clicks the **STOP** button.

5. Display the front panel and run the VI.

   Notice that the VI now runs until you click the **STOP** button. A While Loop executes the functions inside the loop until the user presses the **STOP** button.

Controlling the Speed of Execution

To plot the points on the waveform graph more slowly, you can add a time delay to the block diagram. Complete the following steps to control the speed at which the VI executes.

1. On the block diagram, select the Time Delay Express VI on the **Execution Control** palette and place it inside the loop.
2. Type `.250` in the **Time delay (seconds)** text box.

   This time delay specifies how fast the loop runs. With a .250 second time delay, the loop iterates once every quarter of a second.
3. Close the **Configure Time Delay** dialog box.
4. Save this VI.
5. Display the front panel and run the VI.

6. Click the **Enable** switch and notice the change on the graph.

   Notice how if the **Enable** switch is on, the graph displays the reduced signal. If the **Enable** switch is off, the graph does not display the reduced signal.

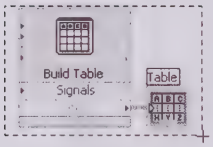
7. Click the **STOP** button to stop the VI.

Using a Table to Display Data

Complete the following steps to display a collection of mean values in a table on the front panel.

1. On the front panel, select the **Express Table** indicator on the **Text Indicators** palette and place it on the front panel to the right of the waveform graph.
2. Display the block diagram.

   Notice that the **Table** terminal appears wired to the Build Table Express VI automatically.



3. If the Build Table Express VI and the **Table** terminal are not selected already, click an open area on the block diagram to the left of the Build Table Express VI and the **Table** terminal. Drag the cursor diagonally until the selection rectangle encloses the Build Table Express VI and the **Table** terminal, shown at left.

   A moving dashed outline called a marquee highlights the Build Table Express VI, the **Table** terminal, and the wire joining the two.

4. Drag the objects into the While Loop to the right of the **Mean** terminal.

   Notice that the While Loop automatically resizes to enclose the Build Table Express VI and the **Table** terminal.

5. Using the Wiring tool, wire the **Mean** terminal of the Sample Compression Express VI to the **Signals** input of the Build Table Express VI.

The block diagram should appear similar to Figure 3-4.

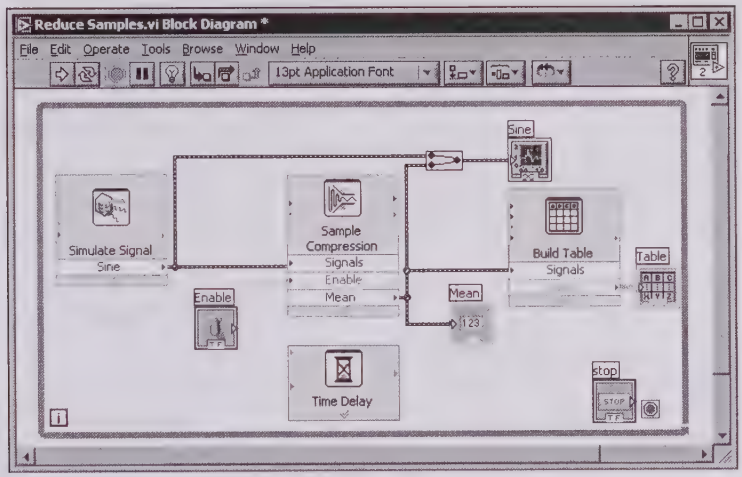


Figure 3-4. Block Diagram of the Reduce Samples VI

6. Display the front panel and run the VI.
7. Click the **Enable** switch.

   The table displays the mean values of every 25 samples of the sine wave. Notice if the **Enable** switch is off, the table does not record the mean values.
8. Stop the VI.
9. Experiment with properties of the table by using the **Table Properties** dialog box. For example, try changing the number of columns to one.
10. Save and close this VI.

Summary

The following topics are a summary of the main concepts you learned in this chapter.

Using the LabVIEW Help Resources

You can use the **Context Help** window and the *LabVIEW Help* to learn more about Express VIs. Both provide information that describe the functionality of the Express VI and how to configure the Express VI.

The following is a summary of the different ways you learned to use the help resources in this chapter.

- The **Context Help** window displays basic information about LabVIEW objects when you move the cursor over each object. Objects with context help information include VIs, structures, palettes, and dialog box components.
- When you place an Express VI on the block diagram, the **Context Help** window displays a brief description of the Express VI and information about how you configured the Express VI.
- You can find and select an Express VI and other block diagram objects in the *LabVIEW Help*. Click the **Place on the block diagram** button to select a block diagram object and place it on the block diagram.
- To navigate the *LabVIEW Help*, use the **Contents**, **Index**, and **Search** tabs. Use the **Contents** tab to get an overview of the topics and structure of the help. Use the **Index** tab to find a topic by keyword. Use the **Search** tab to search the help for a word or phrase.

Customizing the Block Diagram Code

There are many controls, indicators, Express VIs, and structures that you can use to customize a VI. To customize a VI, you can create controls and indicators, control when a VI stops running, and display generated data in a table.

Creating Controls and Indicators

Create controls and indicators that are wired to Express VIs from the block diagram by right-clicking the Express VI input, output, or wire, and selecting an option from the **Create** shortcut menu.

Controlling When a VI Stops Running

Use the While Loop to continually run the code enclosed within the loop. The While Loop stops running when a stop condition is met. When you place or move an object in a While Loop near the border, the loop automatically resizes to add space for that object.

The **Execution Control** palette includes objects that let you control the number of times a VI runs, as well as the speed at which the VI runs.

Displaying Data in a Table

The table indicator displays collected data. Use the Build Table Express VI to build a table of collected data.

4

Acquiring Data and Communicating with Instruments

This chapter introduces you to the Express VIs you use for data acquisition and instrument communication on a PC running Windows.

Refer to the *LabVIEW Measurements Manual* for information about data acquisition and instrument communication on all platforms.

Acquiring a Signal

In the following exercises, you will use the DAQ Assistant Express VI to create an NI-DAQmx task. Refer to the *Taking an NI-DAQmx Measurement in LabVIEW* help tutorial for information about additional methods you can use to create NI-DAQmx tasks. To launch this help tutorial, select **Help»Taking an NI-DAQmx Measurement in LabVIEW**.

Note The following exercises require that you have installed NI-DAQmx and an NI-DAQmx-supported device. Refer to the National Instruments Web site at `ni.com/daq` for the list of NI-DAQmx-supported devices. If you do not have NI-DAQmx installed or an NI-DAQmx-supported device, refer to the *LabVIEW Measurements Manual* for information about using Traditional NI-DAQ for data acquisition.

Complete the following exercises to create an NI-DAQmx task that continuously takes a voltage reading and plots the data on a waveform graph.

You can complete the exercises in approximately 30 minutes.

Creating an NI-DAQmx Task

In NI-DAQmx, a task is a collection of one or more channels, timing, triggering, and other properties that apply to the task itself. Conceptually, a task represents a measurement or generation you want to perform. For example, you can create a task to measure temperature from one or more channels on a DAQ device. Complete the following steps to create and configure a task that reads a voltage level from a DAQ device.

1. Open a new VI.
2. Select the DAQ Assistant Express VI, shown at left, on the **Input** palette and place it on the block diagram. The DAQ Assistant launches and a **Create New** dialog box appears.
3. Click the **Analog Input** button to display the **Analog Input** options.
4. Select **Voltage** to create a new voltage analog input task.

   The dialog box displays a list of channels on each DAQ device installed. The number of channels listed depends on the number of channels you have on the DAQ device.
5. In the **My Physical Channels** listbox, select the physical channel to which the signal is connected, such as **ai0**, and then click the **Finish** button. The DAQ Assistant opens a new window, shown in Figure 4-1, which displays options for configuring the channel you selected to complete a task.

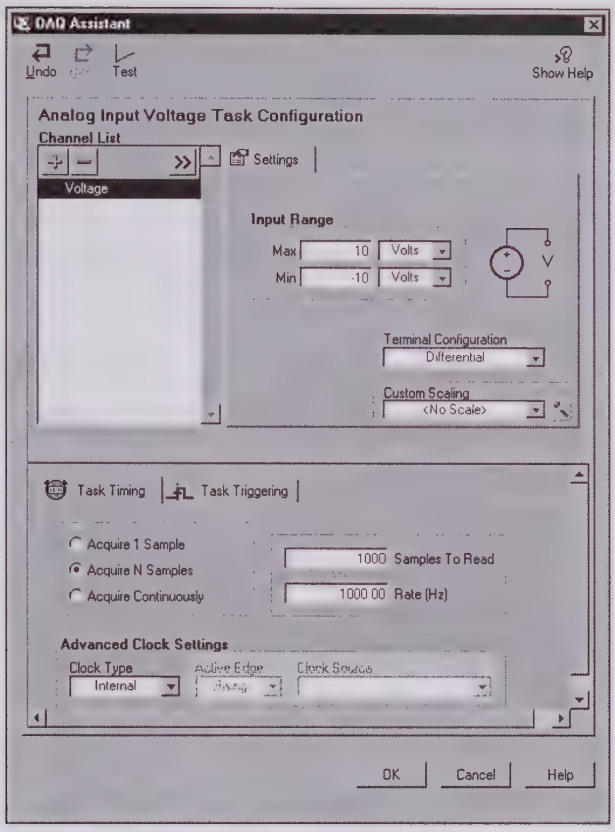


Figure 4-1. Configuring a Task Using the DAQ Assistant

6. In the **Input Range** section of the **Settings** tab, enter `10` for the **Max** value and enter `-10` for the **Min** value.
7. In the **Task Timing** tab, select the **Acquire N Samples** option.
8. Enter a value of `1000` in the **Samples To Read** input box.

Testing the Task

You can test the task to verify that you correctly configured the channel. Complete the following steps to confirm that you are acquiring data.

1. Click the **Test** button, shown at left. An **Analog Input Test Panel** dialog box appears.
2. Click the **Start** button once or twice to confirm that you are acquiring data, and then click the **OK** button to return to the DAQ Assistant.
3. Click the **OK** button to return to the block diagram.
4. Save this VI as `Read Voltage.vi` to an easily accessible location.

Graphing Data from a DAQ Device

Using the task you created in the previous exercise, you can graph the data acquired from a DAQ device. Complete the following steps to plot the data from the channel in a waveform graph and change the name of the signal.

1. On the block diagram, right-click the **data** output and select **Create» Graph Indicator**.
2. Display the front panel.
3. Run the VI three or four times and observe the waveform graph.

   Notice that **`Voltage`** appears in the waveform graph plot legend.
4. Display the block diagram.
5. Right-click the DAQ Assistant Express VI and select **Properties** to rename the channel.
6. Right-click **Voltage** in the **Channel List** listbox and select **Rename** to display the **Rename a channel or channels** dialog box.

Tip You also can select the name of the channel and press the <F2> key to access the **Rename a channel or channels** dialog box.

7. In the **New Name** text box, enter `First Voltage Reading`, and click the **OK** button.
8. Click the **OK** button to apply this configuration and return to the block diagram.
9. Display the front panel and run the VI.

   Notice that **First Voltage Reading** appears in the waveform graph plot legend.
10. Save this VI.

Editing an NI-DAQmx Task

You can add a channel to the task so that you can compare two separate voltage readings. You also can customize the task to acquire the voltage readings continuously. Complete the following steps to add a new channel to the task and acquire data continuously.

1. Display the block diagram and double-click the DAQ Assistant Express VI to add a new channel.
2. Click the **Add Step** button, shown at left, to open the **Add Channels To Task** dialog box.
3. Select any unused physical channel in the **My Physical Channels** listbox.
4. Click the **OK** button to return to the DAQ Assistant.
5. Rename the channel `Second Voltage Reading`.
6. In the **Task Timing** tab, select the **Acquire Continuously** option.

   When you set timing and triggering options in the DAQ Assistant, these options apply to all the channels in the **Channel List**.
7. Click the **OK** button to apply this configuration and return to the block diagram.
8. Place a While Loop around the DAQ Assistant Express VI and the graph indicator wired to the **data** output. The block diagram should appear similar to Figure 4-2.

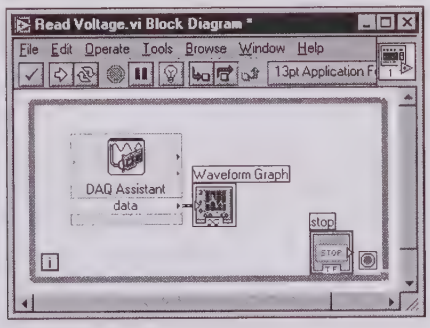


Figure 4-2. Block Diagram for the Read Voltage VI

Visually Comparing Two Voltage Readings

Because you have two voltage readings displayed on a graph, you can customize the plots to distinguish between the two. Complete the following steps to customize the plot color on the waveform graph.

1. On the front panel, expand the plot legend to include two plots.
2. Run the VI.

   Notice two plots now appear in the graph and the plot legend automatically updates to include both plot names.
3. Right-click **First Voltage Reading** and select **Color** from the shortcut menu. Using the color picker, select a color, such as yellow, so that the plot is easy to read. Change the plot color of **Second Voltage Reading**.
4. Save this VI.

Communicating with an Instrument

Instrument drivers simplify instrument control and reduce test program development time by eliminating the need to learn the programming protocol for each instrument. Use an instrument driver for instrument control when possible. National Instruments provides instrument drivers for a wide variety of instruments. Visit the NI Instrument Driver Network at `ni.com/idnet` to find a driver for your instrument.

If a driver is not available for your instrument, you can use the Instrument I/O Assistant Express VI to communicate with your instrument. Complete the following exercises to communicate with an instrument.

Selecting an Instrument

Before communicating with an instrument, you must select the instrument with which you want to communicate. Complete the following steps to select an instrument using the Instrument I/O Assistant Express VI.

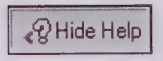
1. Make sure you turn on the instrument you want to use. The instrument must be powered on to use the Instrument I/O Assistant Express VI.
2. Select the Instrument I/O Assistant Express VI on the **Input** palette and place it on the block diagram.

3. Click the **Show Help** button, shown at left, in the upper right corner of the **Instrument I/O Assistant** dialog box.

   Notice how the **Show Help** button displays the help to the right of the dialog box. The help window on top contains procedural information about using the Instrument I/O Assistant. The help window below

provides context-sensitive help about various controls and indicators in the dialog box.

4. Follow the procedures in the help window on top to select the instrument with which you want to communicate.
5. If necessary, configure the properties of the instrument.



6. Click the **Hide Help** button, shown at left, in the upper right corner of the **Instrument I/O Assistant** dialog box to minimize the help window.

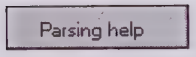
Acquiring and Parsing Information for an Instrument

After selecting the instrument, you can send commands to the instrument to retrieve data. In this exercise, you will learn to use the Instrument I/O Assistant Express VI to acquire and parse identification information for an instrument. Complete the following steps to communicate with your instrument.

1. Click the **Add Step** button, and select **Query and Parse**.
2. Enter `*IDN?` in the **Enter a command** text box.

   `*IDN?` is a query that most instruments recognize. The response is an identification number string that describes the instrument. If the instrument does not accept this command, refer to the reference manual for the instrument for a list of commands the instrument does understand.
3. Click the **Run Sequence** button.

   The Instrument I/O Assistant sends the command to the instrument, and the instrument returns its identification information.
4. Parse the instrument name as an ASCII string. You also can use Instrument I/O Assistant to parse ASCII numbers and binary data.



5. Click the **Parsing help** button, shown at left, in the **Instrument I/O Assistant** dialog box for more information about parsing data.
6. Assign a name to the token in the **Token name** text box.

   A token is a parsed data selection.

7. Click the **OK** button to return to the block diagram.

   Notice that the name that you entered in the **Token name** text box is the output of the Instrument I/O Assistant Express VI, shown at left.

Summary

The following topics are a summary of the main concepts you learned in this chapter.

DAQ Assistant Express VI

You can use the DAQ Assistant Express VI to graphically configure channels or common measurement tasks. Using the DAQ Assistant Express VI, you can interactively build a measurement channel or task.

Place the DAQ Assistant Express VI on the block diagram to configure channels and tasks for use with NI-DAQmx for data acquisition. NI-DAQmx is a programming interface for communicating with data acquisition devices. You can use the DAQ Assistant Express VI to control devices supported in NI-DAQmx.

Refer to the *Taking an NI-DAQmx Measurement in LabVIEW* help tutorial for information about the DAQ Assistant. To launch this help tutorial, select **Help»Taking an NI-DAQmx Measurement in LabVIEW**.

Refer to the National Instruments Web site at `ni.com/daq` for information about devices supported in NI-DAQmx. If your device is not supported in NI-DAQmx, refer to the *LabVIEW Measurements Manual* for information on using Traditional NI-DAQ for data acquisition.

Tasks

In NI-DAQmx, a task is a collection of one or more channels, timing, triggering, and other properties that apply to the task itself. Conceptually, a task represents a measurement or generation you want to perform.

For example, you can configure a collection of channels for analog input operations. After you create a task, you do not have to configure the channels individually to perform analog input operations but instead access the single task. After you create a task, you can add or remove channels from that task.

Refer to the *Channels Versus Tasks* section of Chapter 5, *Creating a Typical Measurement Application*, of the *LabVIEW Measurements Manual* for more information on channels and tasks.

Instrument I/O Assistant Express VI

An instrument driver is a set of software routines that control a programmable instrument. Each routine corresponds to a programmatic operation such as configuring, reading from, writing to, and triggering the instrument. National Instruments offers thousands of instrument drivers online. Visit the NI Instrument Driver Network at `ni.com/idnet` to find a driver for your instrument.

If a driver is not available for your instrument, you can use the Instrument I/O Assistant Express VI to communicate with your instrument. You can use the Instrument I/O Assistant to communicate with a serial, Ethernet, or GPIB instrument and graphically parse the response. Start the Instrument I/O Assistant by placing the Instrument I/O Assistant Express VI on the block diagram or by double-clicking the Instrument I/O Assistant Express VI icon on the block diagram.

Refer to the *Instrument I/O Assistant Help* for information about communicating with an external device.

5

Using Other LabVIEW Features

The previous chapters in this manual introduce you to most of the LabVIEW features you need to build common measurement applications. As you familiarize yourself with the LabVIEW environment, you might find that you need to enhance your VIs or that you need more fine-grain control of the processes your VIs perform. This chapter introduces you to some of the concepts you should be familiar with as you start using other LabVIEW features. Refer to the *LabVIEW User Manual* for more information about these concepts.

NI Example Finder

The **New** dialog box contains many LabVIEW template VIs that you can use to start building VIs. However, these template VIs are only a subset of the hundreds of example VIs included with LabVIEW. You can modify any example VI to fit an application, or you can copy and paste from an example into a VI that you create.

In addition to the example VIs that ship with LabVIEW, you also can access hundreds of example VIs on the NI Developer Zone at `ni.com/zone`. To search all examples using LabVIEW VIs, use the NI Example Finder. The NI Example Finder is the gateway to all installed examples and the examples located on the NI Developer Zone.

To launch the NI Example Finder, select **Help»Find Examples** from the front panel or block diagram menu bar. You also can launch the NI Example Finder by clicking the arrow on the **Open** button on the LabVIEW dialog box and selecting **Examples** from the shortcut menu.

All Controls and Indicators

The controls and indicators located on the subpalettes at the top level of the **Controls** palette are a subset of the complete set of built-in controls and indicators available in LabVIEW. The **All Controls** subpalette contains all the controls and indicators that you can use the create the front panel. However, the **All Controls** subpalette categorizes controls and indicators

by functionality instead of having a subpalette for controls and a subpalette for indicators.

For example, the top level of the **Controls** palette has a **Numeric Controls** subpalette and a **Numeric Indicators** subpalette. On the **All Controls** subpalette, because these controls and indicators are all numeric objects, they are located on the **Numeric** subpalette.

Complete the following steps to change the palette view and display the subpalettes from the **All Controls** subpalette at the top level of the **Controls** palette.

1. Click the **Options** button, shown at left, on the **Controls** palette toolbar to display the **Controls/Functions Palettes** page of the **Options** dialog box.
2. Select **Advanced** from the **Palette View** pull-down menu.
3. Click the **OK** button to apply the changes.

Refer to Chapter 4, *Building the Front Panel*, of the *LabVIEW User Manual* for more information about using the complete set of built-in controls and indicators available in LabVIEW.

All VIs and Functions

The Express VIs and structures located on the subpalettes at the top level of the **Functions** palette are a subset of the complete set of built-in VIs, functions, and structures available in LabVIEW. The **All Functions** subpalette contains all the VIs, functions, and structures that you can use the create the block diagram.

Structures located on the **All Functions** subpalette are similar to the structures located on the **Execution Control** subpalette. For example, when you select a While Loop on the **Execution Control** subpalette and place it on the block diagram, a stop button also appears on the block diagram and is wired to the conditional terminal. However, if you select a While Loop on the **Structures** subpalette and place it on the block diagram, a stop button does not appear.

Complete the following steps to change the palette view and display the subpalettes from the **All Functions** subpalette at the top level of the **Functions** palette.

Note If you change the palette view as described in the *All Controls and Indicators* section of this chapter, the subpalettes from the **All Functions** subpalette appear at the top level of

the **Functions** palette. You do not need to complete the following steps to change the palette view.

1. Click the **Options** button, shown at left, on the **Functions** palette toolbar to display the **Controls/Functions Palettes** page of the **Options** dialog box.
2. Select **Advanced** from the **Palette View** pull-down menu.
3. Click the **OK** button to apply the changes.

LabVIEW uses colored icons to distinguish between VIs, functions, and Express VIs. Icons for VIs have white backgrounds, icons for functions have pale yellow backgrounds, and icons for Express VIs appear on the block diagram as expandable nodes with icons surrounded by a blue field.

Unlike Express VIs, most functions and VIs on the block diagram appear as icons that you cannot expand by default.

VIs

When you place a VI on the block diagram, LabVIEW considers the VI to be a subVI. When you double-click a subVI, a front panel and block diagram appear, rather than a dialog box in which you can configure options. The front panel includes controls and indicators that might look familiar. The block diagram includes wires, front panel icons, functions, possibly subVIs, and other LabVIEW objects that also might look familiar.

The upper right corner of the front panel and block diagram displays the icon for the VI. This icon is the same as the icon that appears when you place the VI on the block diagram.

You can create a VI to use as a subVI. Refer to Chapter 7, *Creating VIs and SubVIs*, of the *LabVIEW User Manual* for more information about creating VIs and configuring them as subVIs.

You also can save the configuration of an Express VI as a subVI. Refer to Chapter 5, *Building the Block Diagram*, of the *LabVIEW User Manual* for more information about creating subVIs from Express VIs.

Functions

Functions are the fundamental operating elements of LabVIEW. Functions do not have front panels or block diagrams but do have connector panes. Double-clicking a function only selects the function.

Data Types

As you build the block diagram for a VI, notice that the terminals for the front panel objects are different colors. The color and symbol of each terminal indicate the data type of the control or indicator. The colors also indicate the data types of the wires, inputs, and outputs. The color of the inputs and outputs on Express VIs indicate what type of data the input or output accepts or returns.

Control terminals have a thicker border than indicator terminals. Also, arrows appear on front panel terminals to indicate whether the terminal is a control or an indicator. An arrow appears on the right if the terminal is a control, and an arrow appears on the left if the terminal is an indicator.

Data types indicate what objects, inputs, and outputs you can wire together. For example, a switch has a green border so you can wire a switch to any input with a green label on an Express VI. A knob has an orange border so you can wire a knob to any input with an orange label. However, you cannot wire a knob to an input with a green label. Notice the wires are of the same color as the terminal.

Refer to Chapter 5, *Building the Block Diagram*, of the *LabVIEW User Manual* for more information about data types.

Dynamic Data Type

The dynamic data type stores the information generated or acquired by an Express VI. The dynamic data type appears as a dark blue terminal, shown at left. Most Express VIs accept and/or return the dynamic data type. You can wire the dynamic data type to any indicator or input that accepts numeric, waveform, or Boolean data. Wire the dynamic data type to an indicator that can best present the data. Indicators include a graph, chart, or numeric indicator.

Most other VIs and functions in LabVIEW do not accept the dynamic data type. To use a built-in VI or function to analyze or process the data the dynamic data type includes, you must convert the dynamic data type.

Converting from Dynamic Data

Use the Convert from Dynamic Data Express VI to convert the dynamic data type to numeric, Boolean, waveform, and array data types for use with other VIs and functions. When you place the Convert from Dynamic Data Express VI on the block diagram, the **Configure Convert from Dynamic**

Data dialog box appears. The **Configure Convert from Dynamic Data** dialog box displays options that let you specify how you want to format the data that the Convert from Dynamic Data Express VI returns.

For example, if you acquired a sine wave from a data acquisition device, select the **Single waveform** option on the **Configure Convert from Dynamic Data** dialog box. Wire the **Waveform** output of the Convert from Dynamic Data Express VI to a function or VI that accepts the waveform data type. If you acquired a collection of temperatures from different channels using a DAQ device, select the **1D array of scalars – most recent values** and **Floating point numbers (double)** options. Then wire the **Array** output of the Convert from Dynamic Data Express VI to a function or VI that accepts a numeric array as an input.

When you wire a dynamic data type to an array indicator, LabVIEW automatically places the Convert from Dynamic Data Express VI on the block diagram. Double-click the Convert from Dynamic Data Express VI to open the **Configure Convert from Dynamic Data** dialog box to control how the data appears in the array.

Converting to Dynamic Data

Use the Convert to Dynamic Data Express VI to convert numeric, Boolean, waveform, and array data types to the dynamic data type for use with Express VIs. When you place the Convert to Dynamic Data Express VI on the block diagram, the **Configure Convert to Dynamic Data** dialog box appears. Use this dialog box to select the kind of data to convert to the dynamic data type.

For example, if you acquire a sine wave using the Analog Input VIs and want to use the Signal Analysis Express VIs to analyze the signal, select the **Single waveform** option on the **Configure Convert to Dynamic Data** dialog box. Then wire the **Dynamic Data Type** output to an Express VI that accepts the dynamic data type as an input.

When to Use Other LabVIEW Features

The Express VIs, structures, and controls and indicators located on the subpalettes at the top level of the **Functions** and **Controls** palettes provide the functionality you need to build common measurement applications. The following list describes the applications that require you to use the VIs, functions, structures, controls, and indicators located on the **All Functions** and **All Controls** subpalettes.

1. **Programatically controlling properties and methods for the LabVIEW environment, VIs, and controls and indicators**—You can programatically control how a VI behaves when it is called or run, set the appearance of a control or indicator, or control how the LabVIEW environment behaves. Refer to Chapter 17, *Programatically Controlling VIs*, of the *LabVIEW User Manual* for more information about these features.
2. **Call code from text-based languages**—You can use LabVIEW to communicate with applications written in a text-based programming language, such as C or C++. Refer to Chapter 20, *Calling Code from Text-Based Programming Languages*, of the *LabVIEW User Manual* for more information about these features.
3. **Communicate with other LabVIEW applications across a network**—You can call a VI that resides on another computer running LabVIEW. Refer to Chapter 18, *Networking in LabVIEW*, of the *LabVIEW User Manual* for more information about these features.
4. **Publish VIs on the Web**—You can publish the front panel of any VI you build to the Web, where users can interact with the front panel. Refer to Chapter 18, *Networking in LabVIEW*, of the *LabVIEW User Manual* for more information about these features.
5. **Save data to a variety of file formats**—You can create files, in addition to the LabVIEW measurement data file format, that other applications can use. Refer to Chapter 14, *File I/O*, of the *LabVIEW User Manual* for more information about these features.
6. **Customize Menus**—You can configure which menu items appear when a user runs a VI. You also can create custom menus. Refer to Chapter 16, *Customizing VIs*, of the *LabVIEW User Manual* for more information about these features.

A

Technical Support and Professional Services

Visit the following sections of the National Instruments Web site at ni.com for technical support and professional services:

- **Support**—Online technical support resources include the following:
  - **Self-Help Resources**—For immediate answers and solutions, visit our extensive library of technical support resources available in English, Japanese, and Spanish at ni.com/support. These resources are available for most products at no cost to registered users and include software drivers and updates, a KnowledgeBase, product manuals, step-by-step troubleshooting wizards, conformity documentation, example code, tutorials and application notes, instrument drivers, discussion forums, a measurement glossary, and so on.
  - **Assisted Support Options**—Contact NI engineers and other measurement and automation professionals by visiting ni.com/support. Our online system helps you define your question and connects you to the experts by phone, discussion forum, or email.
- **Training**—Visit ni.com/custed for self-paced tutorials, videos, and interactive CDs. You also can register for instructor-led, hands-on courses at locations around the world.
- **System Integration**—If you have time constraints, limited in-house technical resources, or other project challenges, NI Alliance Program members can help. To learn more, call your local NI office or visit ni.com/alliance.

If you searched ni.com and could not find the answers you need, contact your local office or NI corporate headquarters. Phone numbers for our worldwide offices are listed at the front of this manual. You also can visit the Worldwide Offices section of ni.com/niglobal to access the branch office Web sites, which provide up-to-date contact information, support phone numbers, email addresses, and current events.

Glossary

A

autoscaling	Ability of scales to adjust to the range of plotted values. On graph scales, autoscaling determines maximum and minimum scale values.

B

block diagram	Pictorial description or representation of a program or algorithm. The block diagram consists of executable icons called nodes and wires that carry data between the nodes. The block diagram is the source code for the VI. The block diagram resides in the block diagram window of the VI.
Boolean controls and indicators	Front panel objects to manipulate and display Boolean (TRUE or FALSE) data.
broken **Run** button	Button that replaces the **Run** button when a VI cannot run because of errors.
broken VI	VI that cannot run because of errors; signified by a broken arrow in the broken **Run** button.

C

channel

1. Physical—a terminal or pin at which you can measure or generate an analog or digital signal. A single physical channel can include more than one terminal, as in the case of a differential analog input channel or a digital port of eight lines. The name used for a counter physical channel is an exception because that physical channel name is not the name of the terminal where the counter measures or generates the digital signal.

2. Virtual—a collection of property settings that can include a name, a physical channel, input terminal connections, the type of measurement or generation, and scaling information. You can define NI-DAQmx virtual channels outside a task (global) or inside a task (local). Configuring virtual channels is optional in Traditional NI-DAQ and earlier versions, but is integral to every measurement you take in NI-DAQmx. In Traditional NI-DAQ, you configure virtual channels in MAX. In NI-DAQmx, you can configure virtual channels either in MAX or in a program, and you can configure channels as part of a task or separately.

3. Switch—a switch channel represents any connection point on a switch. It may be made up of one or more signal wires (commonly one, two, or four), depending on the switch topology. A virtual channel cannot be created with a switch channel. Switch channels may be used only in the NI-DAQmx Switch functions and VIs.

checkbox

Small square box in a dialog box you can place or remove checkmarks. Checkboxes generally are associated with multiple options that you can set. You can place a checkmark in more than one checkbox.

conditional terminal

Terminal of a While Loop that contains a Boolean value that determines if the VI performs another iteration.

Context Help window

Window that displays basic information about LabVIEW objects when you move the cursor over each object. Objects with context help information include VIs, functions, constants, structures, palettes, properties, methods, events, and dialog box components.

control

Front panel object for entering data to a VI interactively or to a subVI programmatically, such as a knob, push button, or dial.

Controls palette

Palette that contains front panel controls, indicators, and decorative objects.

current VI

VI whose front panel, block diagram, or Icon Editor is the active window.

D

DAQ	*See* data acquisition (DAQ).
DAQ Assistant	A graphical interface for configuring measurement tasks, channels, and scales.
DAQ device	A device that acquires or generates data and can contain multiple channels and conversion devices. DAQ devices include plug-in devices, PCMCIA cards, and DAQPad devices, which connect to a computer USB or 1394 (FireWire) port. SCXI modules are considered DAQ devices.
data acquisition (DAQ)	1. Acquiring and measuring analog or digital electrical signals from sensors, acquisition transducers, and test probes or fixtures. 2. Generating analog or digital electrical signals.
data flow	Programming system that consists of executable nodes that execute only when they receive all required input data and produce output automatically when they execute. LabVIEW is a dataflow system.
datalog	To acquire data and simultaneously store it in a disk file. LabVIEW file I/O VIs and functions can log data.
data type	Format for information. In LabVIEW, acceptable data types for most VIs and functions are numeric, array, string, Boolean, path, refnum, enumeration, waveform, and cluster.
default	Preset value. Many VI inputs use a default value if you do not specify a value.
device	An instrument or controller you can access as a single entity that controls or monitors real-world I/O points. A device often is connected to a host computer through some type of communication network. *See also* DAQ device and measurement device.
drag	To use the cursor on the screen to select, move, copy, or delete objects.
driver	Software unique to the device or type of device that includes the set of commands the device accepts.

E

error message	Indication of a software or hardware malfunction or of an unacceptable data entry attempt.
Express VI	A subVI designed to aid in common measurement tasks. You configure an Express VI using a configuration dialog box.

F

front panel	Interactive user interface of a VI. Front panel designs imitate physical instruments, such as oscilloscopes and multimeters.
function	Built-in execution element, comparable to an operator, function, or statement in a text-based programming language.
Functions palette	Palette that contains VIs, functions, block diagram structures, and constants.

G

graph	2D display of one or more plots. A graph receives and plots data as a block.

I

I/O	Input/Output. The transfer of data to or from a computer system involving communications channels, operator input devices, and/or data acquisition and control interfaces.
icon	Graphical representation of a node on a block diagram.
indicator	Front panel object that displays output, such as a graph or LED.
instrument driver	A set of high-level functions that control and communicate with instrument hardware in a system.
iteration terminal	Terminal of a For Loop or While Loop that contains the current number of completed iterations.

L

label — Text object used to name or describe objects or regions on the front panel or block diagram.

LabVIEW — Laboratory Virtual Instrument Engineering Workbench. LabVIEW is a graphical programming language that uses icons instead of lines of text to create programs.

LED — Light-emitting diode.

legend — Object a graph or chart owns to displays the names and plot styles of plots on that graph or chart.

listbox — Box within a dialog box that lists all available choices for a command. For example, a list of filenames on a disk.

M

measurement device — DAQ devices such as the E Series multifunction I/O (MIO) devices, SCXI signal conditioning modules, and switch modules.

menu bar — Horizontal bar that lists the names of the main menus of an application. The menu bar appears below the titlebar of a window. Each application has a menu bar that is distinct for that application, although some menus and commands are common to many applications.

N

NI-DAQ — Driver software included with all NI measurement devices. NI-DAQ is an extensive library of VIs and functions you can call from an application development environment (ADE), such as LabVIEW, to program all the features of an NI measurement device, such as configuring, acquiring and generating data from, and sending data to the device.

NI-DAQmx	The latest NI-DAQ driver with new VIs, functions, and development tools for controlling measurement devices. The advantages of NI-DAQmx over earlier versions of NI-DAQ include the DAQ Assistant for configuring channels and measurement tasks for your device for use in LabVIEW, LabWindows™/CVI™, and Measurement Studio; increased performance such as faster single-point analog I/O; and a simpler API for creating DAQ applications using fewer functions and VIs than earlier versions of NI-DAQ.
node	Program execution element. Nodes are analogous to statements, operators, functions, and subroutines in text-based programming languages. On a block diagram, nodes include functions, structures, and subVIs.
numeric controls and indicators	Front panel objects to manipulate and display numeric data.

O

object	Generic term for any item on the front panel or block diagram, including controls, indicators, structures, nodes, wires, and imported pictures.
Operating tool	Tool to enter data into controls operate them.

P

palette	Displays objects or tools you can use to build the front panel or block diagram.
panel window	VI window that contains the front panel, the toolbar, and the icon and connector panes.
plot	Graphical representation of an array of data shown either on a graph or a chart.
Positioning tool	Tool to move and resize objects.
pull-down menus	Menus accessed from a menu bar. Pull-down menu items are usually general in nature.

S

sample	Single analog or digital input or output data point.
scale	Part of graph, chart, and some numeric controls and indicators that contains a series of marks or points at known intervals to denote units of measure.
shortcut menu	Menu accessed by right-clicking an object. Menu items pertain to that object specifically.
string	Representation of a value as text.
structure	Program control element, such as a Flat Sequence structure, Stacked Sequence structure, Case structure, For Loop, or While Loop.
subVI	VI used on the block diagram of another VI. Comparable to a subroutine.

T

task	A collection of one or more channels, timing, triggering, and other properties in NI-DAQmx. A task represents a measurement or generation you want to perform.
terminal	Object or region on a node through which data pass.
tip strip	Small yellow text banners that identify the terminal name and make it easier to identify terminals for wiring.
tool	Special cursor to perform specific operations.
toolbar	Bar that contains command buttons to run and debug VIs.
Traditional NI-DAQ	An upgrade to the earlier version of NI-DAQ. Traditional NI-DAQ has the same VIs and functions and works the same way as NI-DAQ 6.9.*x*. You can use both Traditional NI-DAQ and NI-DAQmx on the same computer, which is not possible with NI-DAQ 6.9.*x*.

U

URL	Uniform resource locator. A logical address that identifies a resource on a server, usually on the Web. For example, `http://www.ni.com/` is the URL for the National Instruments Web site.

V

VI	*See* virtual instrument (VI).
virtual instrument (VI)	Program in LabVIEW that models the appearance and function of a physical instrument.

W

waveform	Multiple voltage readings taken at a specific sampling rate.
waveform chart	Indicator that plots data points at a certain rate.
While Loop	Loop structure that repeats a section of code until a condition is met.
wire	Data path between nodes.
wire branch	Section of wire that contains all the wire segments from junction to junction, terminal to junction, or terminal to terminal if there are no junctions between.
wire junction	Point where three or more wire segments join.
wire segment	Single horizontal or vertical piece of wire.
Wiring tool	Tool to define data paths between terminals.

Index

Symbols

.lvm file, 2-9, 2-14

A

Abort Execution button, 1-8
acquiring
    information for instruments, 4-7
    signals, 4-1
adding
    controls, 3-4
    controls to front panel, 1-4
    graph indicator, 3-4
    inputs to Express VIs, 1-6, 2-11, 3-4
    new channels to tasks, 4-5
    numeric indicators, 3-4
    outputs to Express VIs, 2-11
    visual cues on front panel, 2-6
    warning light, 2-6
All Controls palette, 5-1
    changing palette view, 5-2
All Functions palette, 5-2
    changing palette view, 5-2
Amplitude and Level Measurements Express VI, 2-2
    analyzing voltage, 2-5
analyzing
    signals, 2-5
Arithmetic & Comparison palette, 1-9

B

block diagram, 1-4, 1-17
    indicators, 2-13
    modifying, 2-3
    show, 1-5
broken
    Run button, 2-4
    wires, 2-4, 2-14
Build Table Express VI, 3-7
button
    Abort Execution, 1-8
    place on block diagram, 3-9
    Run, 1-4

C

calling code from text-based languages, 5-6
changing
    palette views, 5-2
    signal type, 1-5
channels, 4-2
    adding to tasks, 4-5
    renaming, 4-4
communicating
    with instruments, 4-6, 4-9
    with LabVIEW applications across networks, 5-6
Comparison Express VI
    setting warning level, 2-7
configuration dialog box, 1-17
contacting National Instruments, A-1
Context Help window, 2-2, 2-13, 3-2, 3-9
    button, 3-2
    display configuration of Express VI, 2-2
    displaying errors, 2-14
controlling
    execution speed, 3-6, 3-10
controls, 1-16, 2-13
    adding, 3-4
    adding to front panel, 1-4
    all, 5-1
    configuring, 1-16

creating, 3-4, 3-9
customizing, 1-12
data type, 5-4
numeric, 1-5
palette, 1-4
Controls palette, 1-4
changing palette views, 5-2
conventions used in this manual, *ix*
Convert from Dynamic Data Type Express VI, 5-4
Convert to Dynamic Data Type Express VI, 5-5
creating
controls, 3-4, 3-9
graph indicator, 3-4
indicators, 3-9
NI-DAQmx tasks, 4-2
customer
education, A-1
professional services, A-1
technical support, A-1
customizing
block diagrams, 3-9
controls, 1-12
front panel, 3-4
indicators, 1-14
menus, 5-6
simulated signal, 2-3

D

DAQ Assistant Express VI, 4-2, 4-8
DAQ device, 4-2
data flow, 1-7, 1-11, 1-17
data type
dynamic, 5-4
data types, 5-4
deleting
wires, 1-9
deselecting objects, 1-7
diagnostic resources, A-1
displaying
data from DAQ devices, 4-4
data in tables, 3-7, 3-10
signals in a graph, 1-11
documentation
conventions used in this manual, *ix*
introduction to this manual, *ix*
online library, A-1
drivers, 4-6, 4-9
instrument, 4-6, 4-9, A-1
software, A-1
dynamic data type, 5-4
converting from, 5-4
converting to, 5-5

E

Error list window, 2-4, 2-14
errors, 2-14
displaying, 2-4
displaying in Context Help window, 2-14
list, 2-4, 2-14
window, 2-4, 2-14
example code, A-1
example VIs
NI Example Finder, 5-1
Execution Control palette, 3-5, 3-10
execution speed
controlling, 3-6
Express VI
Convert to Dynamic Data Type, 5-5
Express VIs, 1-17
Amplitude and Level Measurements, 2-2, 2-5
Build Table, 3-7
Comparison, 2-7
configuration dialog box, 1-17
Convert from Dyanmic Data Type, 5-4
DAQ Assistant, 4-2, 4-8
dynamic data type, 5-4
inputs, 1-17

Instrument I/O Assistant, 4-6, 4-9
outputs, 1-17
Sample Compression, 3-3
Scaling and Mapping, 1-9
Simulate Signals, 1-5
Time Delay, 3-6
Write LabVIEW Measurement File, 2-9, 2-14

F

frequently asked questions, A-1
front panel, 1-4, 1-16
adding controls, 1-4
controls, 1-16, 2-13
customizing, 3-4
indicators, 1-16
modifying, 2-4
show, 1-8
functions, 5-3
all, 5-2
Merge Signals, 1-12
merge signals, 3-4
Functions palette, 1-9
changing palette views, 5-2

G

graph indicator
adding, 3-4
creating, 3-4
graphing
data from DAQ devices, 4-4
two signals, 1-11

H

help
button, 2-5
Context Help window, 2-2, 2-13, 3-2, 3-9
LabVIEW Help, 2-5, 2-13, 3-9
LabVIEW resources, 2-13, 3-9
professional services, A-1
technical support, A-1

I

indicators, 1-16, 2-13
adding numeric, 3-4
all, 5-1
configuring, 1-16
creating, 3-9
customizing, 1-14
data type, 5-4
removing, 2-4
Input palette, 3-2
instrument drivers, 4-6, 4-9, A-1
Instrument I/O Assistant Express VI, 4-6, 4-9
instruments
acquiring information, 4-7
communicating, 4-6, 4-9
parsing information, 4-7
selecting, 4-6

K

KnowledgeBase, A-1

L

LabVIEW dialog box, 1-2, 2-2
LabVIEW Help, 2-13, 3-9
searching for Express VI, 3-3
LabVIEW help resources
using, 3-9
LEDs, 2-6
LEDs palette
palette, 2-6

M

manual. *See* documentation
marquee, 3-7
Merge Signals function, 1-12, 3-4
modifying
    block diagram, 2-3
    front panel, 2-4
    signals, 1-9, 3-3

N

National Instruments
    customer education, A-1
    professional services, A-1
    system integration services, A-1
    technical support, A-1
    worldwide offices, A-1
New dialog box, 1-2, 1-16, 2-2
NI Example Finder, 5-1
NI Instrument Driver Network, 4-6, 4-9
NI-DAQmx
    creating tasks, 4-2
    tasks, 4-2, 4-8
    testing tasks, 4-4
Numeric Controls palette, 1-5

O

objects
    deselecting, 1-7
    selecting, 1-7
online technical support, A-1
operating
    tool, 1-8

P

palettes
    All Controls, 5-1
    All Functions, 5-2
    Arithmetic & Comparison, 1-9
    changing palette views, 5-2
    Controls, 1-4
    Execution Control, 3-5, 3-10
    Functions, 1-9
    Input, 3-2
    Numeric Controls, 1-5
    Text Indicators, 3-7
parsing
    information for instruments, 4-7
phone technical support, A-1
place on block diagram, 3-9
positioning
    tool, 1-7
professional services, A-1
programatically controlling VIs, 5-6
programming examples, A-1
property dialog boxes, 1-16
publishing VIs on the Web, 5-6

R

Run button, 1-4
    broken, 2-4, 2-14
    running, 1-8
running VIs, 1-8
    continuously, 3-5

S

Sample Compression Express VI, 3-3
saving data
    to different file formats, 5-6
    to files, 2-9, 2-14
    when prompted by user, 2-10
Scaling and Mapping Express VI, 1-9
    define slope, 1-10
selecting
    instruments, 4-6
    objects, 1-7

signals
acquiring, 4-1
analyzing, 2-5
changing type, 1-5
modifying, 1-9, 3-3
Simulate Signal Express VI, 1-5
sine wave, 1-5
software drivers, A-1
support
technical, A-1
system integration services, A-1

T

table, 3-7
displaying data, 3-10
tasks
adding new channels, 4-5
NI-DAQmx, 4-8
testing, 4-4
technical support, A-1
telephone technical support, A-1
template VIs, 1-2, 1-16
blank, 3-1
terminals, 1-7
Text Indicators palette, 3-7
text-based languages
calling code, 5-6
Time Delay Express VI, 3-6
tools
Operating, 1-8
Positioning, 1-7
Wiring, 1-7
training
customer, A-1
troubleshooting resources, A-1

U

user interface. *See* front panel

V

virtual instruments. *See* VIs
VIs, 1-1
all, 5-2
blank, 3-1, 3-2
building, 1-1, 2-1, 3-1
customizing menus, 5-6
examples, 5-1
icons, 5-3
new, 3-2
programmatically controlling, 5-6
publishing on the Web, 5-6
running, 1-8
subVIs, 5-3
template, 1-2, 1-16

W

Web
professional services, A-1
technical support, A-1
While Loop, 3-6
wires
broken, 2-4, 2-14
deleting, 1-9
wiring
objects on block diagram, 1-7
tool, 1-7
worldwide technical support, A-1
Write LabVIEW Measurement File
saving data, 2-9
Write LabVIEW Measurement File Express VI, 2-9, 2-14

MP MASTER
MA31852